

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

IN RE APPLICATION OF: Shinichiro HAMADA

GAU:

SERIAL NO: New Application

EXAMINER:

FILED: Herewith

FOR: DOCUMENT EDITING METHOD, DOCUMENT EDITING SYSTEM, SERVER APPARATUS, AND  
DOCUMENT EDITING PROGRAM

**REQUEST FOR PRIORITY**

COMMISSIONER FOR PATENTS  
ALEXANDRIA, VIRGINIA 22313

SIR:

☐ Full benefit of the filing date of U.S. Application Serial Number , filed , is claimed pursuant to the provisions of 35 U.S.C. §120.

☐ Full benefit of the filing date(s) of U.S. Provisional Application(s) is claimed pursuant to the provisions of 35 U.S.C. §119(e): Application No. Date Filed

☒ Applicants claim any right to priority from any earlier filed applications to which they may be entitled pursuant to the provisions of 35 U.S.C. §119, as noted below.

In the matter of the above-identified application for patent, notice is hereby given that the applicants claim as priority:

<u>COUNTRY</u>	<u>APPLICATION NUMBER</u>	<u>MONTH/DAY/YEAR</u>
Japan	2002-200284	July 9, 2002
Japan	2002-200285	July 9, 2002

Certified copies of the corresponding Convention Application(s)

☒ are submitted herewith

☐ will be submitted prior to payment of the Final Fee

☐ were filed in prior application Serial No. filed

☐ were submitted to the International Bureau in PCT Application Number

Receipt of the certified copies by the International Bureau in a timely manner under PCT Rule 17.1(a) has been acknowledged as evidenced by the attached PCT/IB/304.

☐ (A) Application Serial No.(s) were filed in prior application Serial No. filed ; and


☐ (B) Application Serial No.(s)

☐ are submitted herewith

☐ will be submitted prior to payment of the Final Fee

Respectfully Submitted,

OBLON, SPIVAK, McCLELLAND,  
MAIER & NEUSTADT, P.C.

  
Marvin J. Spivak

Registration No. 24,913

C. Irvin McClelland  
Registration Number 21,124



22850

0350/39  
1/2

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application: 2002年 7月 9日

出 願 番 号

Application Number: 特願2002-200284

[ST.10/C]:

[JP2002-200284]

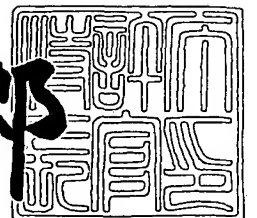
出 願 人

Applicant(s): 株式会社東芝

2003年 2月 7日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

太田信一郎



出証番号 出証特2003-3005078

【書類名】 特許願

【整理番号】 A000203178

【提出日】 平成14年 7月 9日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 7/00

【発明の名称】 文書編集方法、文書編集システム、サーバ装置および文  
書編集プログラム

【請求項の数】 28

【発明者】

【住所又は居所】 神奈川県川崎市幸区小向東芝町1番地 株式会社東芝研  
究開発センター内

【氏名】 浜田 伸一郎

【特許出願人】

【識別番号】 000003078

【氏名又は名称】 株式会社 東芝

【代理人】

【識別番号】 100058479

【弁理士】

【氏名又は名称】 鈴江 武彦

【電話番号】 03-3502-3181

【選任した代理人】

【識別番号】 100084618

【弁理士】

【氏名又は名称】 村松 貞男

【選任した代理人】

【識別番号】 100068814

【弁理士】

【氏名又は名称】 坪井 淳

【選任した代理人】

【識別番号】 100092196

【弁理士】

【氏名又は名称】 橋本 良郎

【選任した代理人】

【識別番号】 100091351

【弁理士】

【氏名又は名称】 河野 哲

【選任した代理人】

【識別番号】 100088683

【弁理士】

【氏名又は名称】 中村 誠

【選任した代理人】

【識別番号】 100070437

【弁理士】

【氏名又は名称】 河井 将次

【手数料の表示】

【予納台帳番号】 011567

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 文書編集方法、文書編集システム、サーバ装置および文書編集プログラム

【特許請求の範囲】

【請求項 1】

複数の要素から構成された文書構造を有する構造化文書を表示手段に表示するとともに、この表示された構造化文書に一致または含まれる、少なくとも 1 つの前記要素からなる予め操作単位として定められた部分文書を基に前記表示された構造化文書を編集する文書編集方法であって、

前記部分文書には、少なくとも当該部分文書が前記操作単位であることを表す情報が含まれており、この情報を基に、少なくとも前記部分文書と、当該部分文書の前記文書構造上の位置情報とを含む前記部分文書毎の部品データを作成して、前記構造化文書を前記部品データの集合として記憶手段に記憶し、

操作対象として選択された前記部分文書に対する操作内容に応じて、当該操作対象の部分文書に対応する前記部品データを更新することにより前記構造化文書を編集することを特徴とする文書編集方法。

【請求項 2】

前記部分文書には、当該部分文書のタイプと、当該部分文書のデータ形式を他のデータ形式に変換するための少なくとも 1 つの変換規則と、当該部分文書として前記表示手段に表示される他の構造化文書中の部分文書へのリンク情報と、当該部分文書に挿入可能な他の部分文書のタイプと挿入位置のうちの少なくとも 1 つが、当該部分文書の付加情報として含まれており、この付加情報が前記部品データにも含まれていることを特徴とする請求項 1 記載の文書編集方法。

【請求項 3】

前記操作対象の前記部分文書の前記文書構造上の位置を移動させるときには、当該部分文書に対応する部品データ中の位置情報を移動先の位置に対応するよう変更することを特徴とする請求項 1 記載の文書編集方法。

【請求項 4】

前記構造化文書としての第 1 の構造化文書と第 2 の構造化文書を前記表示手段

に表示し、前記第 1 の構造化文書に含まれる前記部分文書としての第 1 の部分文書を前記第 2 の構造化文書の指定された位置に挿入するときには、当該第 1 の部分文書に対応する前記部品データとしての第 1 の部品データ中の位置情報を前記第 2 の構造化文書中の挿入位置に対応するよう変更した結果得られた第 2 の部品データを当該第 2 の構造化文書の部品データとして前記記憶手段に記憶することを特徴とする請求項 1 記載の文書編集方法。

【請求項 5】

前記構造化文書としての第 1 の構造化文書と第 2 の構造化文書を前記表示手段に表示し、前記第 1 の構造化文書に含まれる前記部分文書としての第 1 の部分文書を前記第 2 の構造化文書の指定された位置に挿入するときは、

少なくとも前記第 2 の構造化文書中の指定位置と前記第 1 の部分文書に対応する部品文書とを関連付けるリンク情報を含む新たな部品データ作成して、これを前記第 2 の構造化文書の部品データとして前記記憶手段に記憶することを特徴とする請求項 1 記載の文書編集方法。

【請求項 6】

前記構造化文書としての第 1 の構造化文書と第 2 の構造化文書を前記表示手段に表示して、前記第 1 の構造化文書に含まれる前記部分文書としての第 1 の部分文書を前記第 2 の構造化文書に含まれる前記部分文書としての第 2 の部分文書に挿入するときには、

当該第 1 の部分文書に対応する前記部品データとしての第 1 の部品データ中の当該第 1 の部分文書のタイプと、当該第 2 の部分文書に対応する前記部品データとしての第 2 の部品データ中の当該第 2 の部分文書に挿入可能な他の部分文書のタイプとを基に、前記第 1 の部分文書が前記第 2 の部分文書に挿入可能か否かを判断し、

挿入可能であるときは、前記第 1 の部品データ中の位置情報を前記第 2 の構造化文書中の挿入位置に対応させて変更した結果得られる第 3 の部品データを当該第 2 の構造化文書の部品データとして前記記憶手段に記憶し、

挿入不可であるときは、前記第 1 の部品データ中の前記第 1 の部分文書のデータ形式を前記第 2 の部分文書に挿入可能なデータ形式に変換した後、前記第 1 の

部品データ中の位置情報を前記第2の構造化文書中の挿入位置に対応させて変更した結果得られる新たな部品データを当該第2の構造化文書の部品データとして前記記憶手段に記憶することを特徴とする請求項2記載の文書編集方法。

【請求項7】

前記表示手段に表示されている前記操作対象の部分文書を前記表示手段に表示された他の部分文書に挿入するときには、

前記操作対象の部分文書のデータ形式が、前記他の部分文書に挿入可能なデータ形式であるときは、当該操作対象の部分文書に対応する部品データ中の位置情報を前記他の部分文書中の挿入位置に対応させて変更した結果得られる新たな部品データを当該他の部分文書の部品データとして前記記憶手段に記憶し、

前記操作対象の部分文書のデータ形式が、前記他の部分文書に挿入可能なデータ形式でないときは、当該操作対象の部分文書のデータ形式を当該他の部分文書に挿入可能なデータ形式に変換した後、当該操作対象の部分文書に対応する部品データ中の位置情報を前記他の部分文書中の挿入位置に対応させて変更した結果得られる新たな部品データを当該他の部分文書の部品データとして前記記憶手段に記憶し、

さらに、前記操作対象の部分文書と前記他の部分文書とが同じ1つの部分文書に包含されているときは、前記新たな部品データを前記記憶手段に記憶するとともに、当該操作対象の部分文書に対応する部品データを前記記憶手段から削除することを特徴とする請求項1記載の文書編集方法。

【請求項8】

前記操作対象の部分文書に対応する前記部品データには、複数の前記変換規則が含まれており、それらのうちの1つを用いて、当該操作対象の部分文書の表示形式を切り替えることを特徴とする請求項2記載の文書編集方法。

【請求項9】

前記操作対象の部分文書を削除するときには、当該操作対象の部分文書に対応する部品データと当該操作対象の部分文書に包含される部分文書に対応する部品データとを前記記憶手段から削除することを特徴とする請求項1記載の文書編集方法。

【請求項 10】

前記操作対象の部分文書を編集するときには、当該部分文書に対応する部品データ中の当該部分文書を更新する請求項 1 記載の文書編集方法。

【請求項 11】

前記表示手段に表示された構造化文書に新たな部分文書を追加挿入するときには、少なくとも当該新たな部分文書と、この新たな部分文書の前記文書構造上の挿入位置に対応する位置情報とを含む前記部品データを作成して、前記記憶手段に記憶することを特徴とする請求項 1 記載の文書編集方法。

【請求項 12】

前記表示手段に表示された部分文書の中から、ユーザにより指定された範囲内の部分文書を抽出し、この抽出した部分文書に対応する前記部品データを作成して、前記記憶手段に記憶することを特徴とする請求項 1 記載の文書編集方法。

【請求項 13】

複数の要素から構成された文書構造を有する構造化文書を表示手段に表示するとともに、この表示された構造化文書に一致または含まれる、少なくとも 1 つの前記要素からなる予め操作単位として定められた部分文書を基に前記表示された構造化文書を編集する文書編集システムであって、

前記部分文書には、少なくとも当該部分文書が前記操作単位であることを表す情報が含まれており、この情報を基に、少なくとも前記部分文書と、当該部分文書の前記文書構造上の位置情報とを含む前記部分文書毎の部品データを作成する手段と、

前記構造化文書を前記部品データの集合として記憶する記憶手段と、

操作対象として選択された前記部分文書に対する操作内容に応じて、当該操作対象の部分文書に対応する前記部品データを更新することにより前記構造化文書を編集する手段と、

を具備したことを特徴とする文書編集システム。

【請求項 14】

前記部分文書には、当該部分文書のタイプと、当該部分文書のデータ形式を他のデータ形式に変換するための少なくとも 1 つの変換規則と、当該部分文書とし



て前記表示手段に表示される他の構造化文書中の部分文書へのリンク情報と、当該部分文書に挿入可能な他の部分文書のタイプと挿入位置のうちの少なくとも１つが、当該部分文書の付加情報として含まれており、この付加情報が前記部品データにも含まれていることを特徴とする請求項 1 3 記載の文書編集システム。

【請求項 1 5】

前記操作対象の前記部分文書の前記文書構造上の位置を移動させるときには、当該部分文書に対応する部品データ中の位置情報を移動先の位置に対応するよう変更することを特徴とする請求項 1 3 記載の文書編集システム。

【請求項 1 6】

前記構造化文書としての第 1 の構造化文書と第 2 の構造化文書を前記表示手段に表示して、前記第 1 の構造化文書に含まれる前記部分文書としての第 1 の部分文書を前記第 2 の構造化文書の指定された位置に挿入するときには、当該第 1 の部分文書に対応する前記部品データとしての第 1 の部品データ中の位置情報を前記第 2 の構造化文書中の挿入位置に対応するよう変更した結果得られた第 2 の部品データを当該第 2 の構造化文書の部品データとして前記記憶手段で記憶することを特徴とする請求項 1 3 記載の文書編集システム。

【請求項 1 7】

前記構造化文書としての第 1 の構造化文書と第 2 の構造化文書を前記表示手段に表示して、前記第 1 の構造化文書に含まれる前記部分文書としての第 1 の部分文書を前記第 2 の構造化文書の指定された位置に挿入するときは、

前記第 2 の構造化文書に挿入する前記第 2 の部分文書に対応する前記部品データとして、少なくとも前記第 2 の構造化文書中の指定位置から前記第 1 の部分文書へ設定されたリンク情報を含む新たな部品データ作成して、これを前記第 2 の構造化文書の部品データとして前記記憶手段で記憶することを特徴とする請求項 1 3 記載の文書編集システム。

【請求項 1 8】

前記表示手段に表示されている前記操作対象の部分文書を前記表示手段に表示されている他の部分文書に挿入するときには、

前記操作対象の部分文書のデータ形式が、前記他の部分文書に挿入可能なデー

タ形式であるときは、当該操作対象の部分文書に対応する部品データ中の位置情報を前記他の部分文書中の挿入位置に対応させて変更した結果得られる新たな部品データを当該他の部分文書の部品データとして前記記憶手段に記憶し、

前記操作対象の部分文書のデータ形式が、前記他の部分文書に挿入可能なデータ形式でないときは、当該操作対象の部分文書のデータ形式を当該他の部分文書に挿入可能なデータ形式に変換した後、当該操作対象の部分文書に対応する部品データ中の位置情報を前記他の部分文書中の挿入位置に対応させて変更した結果得られる新たな部品データを当該他の部分文書の部品データとして前記記憶手段に記憶することを特徴とする請求項 1 3 記載の文書編集システム。

【請求項 1 9】

前記操作対象の部分文書に対応する前記部品データには、複数の前記変換規則が含まれており、それらのうちの 1 つを用いて、当該操作対象の部分文書の表示形式を切り替えることを特徴とする請求項 1 3 記載の文書編集システム。

【請求項 2 0】

前記操作対象の部分文書を削除するときには、当該操作対象の部分文書に対応する部品データと当該操作対象の部分文書に包含される部分文書に対応する部品データとを前記記憶手段から削除することを特徴とする請求項 1 3 記載の文書編集システム。

【請求項 2 1】

前記操作対象の部分文書を編集するときには、当該部分文書に対応する部品データ中の当該部分文書を更新する請求項 1 3 記載の文書編集システム。

【請求項 2 2】

前記表示手段に表示された構造化文書に新たな部分文書を追加挿入するときには、少なくとも当該新たな部分文書と、この新たな部分文書の前記文書構造上の挿入位置に対応する位置情報とを含む前記部品データを作成して、前記記憶手段で記憶することを特徴とする請求項 1 3 記載の文書編集システム。

【請求項 2 3】

前記表示手段に表示された部分文書の中から、ユーザにより指定された範囲内の部分文書を抽出し、この抽出した部分文書に対応する前記部品データを作成し

て、前記記憶手段に記憶することを特徴とする請求項 13 記載の文書編集システム。

【請求項 24】

複数の要素から構成された文書構造を有する構造化文書を表示手段に表示するとともに、この表示された構造化文書に一致または含まれる、少なくとも 1 つの前記要素からなる予め操作単位として定められた部分文書を基に前記表示された構造化文書を編集するクライアント装置に対し、前記構造化文書を配信するサーバ装置であって、

前記クライアント端末に配信する前記構造化文書には、少なくとも当該構造化文書に一致または含まれる前記操作単位としての部分文書を表す情報が含まれることを特徴とするサーバ装置。

【請求項 25】

複数の要素から構成された文書構造を有する構造化文書を表示手段に表示するとともに、この表示された構造化文書に一致または含まれる、少なくとも 1 つの前記要素からなる予め操作単位として定められた部分文書を基に前記表示された構造化文書を編集するクライアント装置に対し、前記構造化文書を配信するサーバ装置であって、

前記クライアント端末に配信する前記構造化文書には、当該構造化文書に一致または含まれる前記操作単位としての部分文書を表す情報とともに、各部分文書について、当該部分文書のタイプと、当該部分文書のデータ形式を他のデータ形式に変換するための少なくとも 1 つの変換規則と、当該部分文書として前記表示手段に表示される他の構造化文書中の部分文書へのリンク情報と、当該部分文書に挿入可能な他の部分文書のタイプおよび挿入位置のうちの少なくとも 1 つが、当該部分文書の付加情報として含まれていることを特徴とするサーバ装置。

【請求項 26】

前記クライアント端末に配信する前記構造化文書に、少なくとも当該構造化文書に一致または含まれる前記操作単位としての部分文書を表す情報が含まれないときは、少なくとも前記操作単位としての部分文書を表す情報を挿入する手段を具備したことを特徴とする請求項 24 または 25 記載のサーバ装置。

【請求項 2 7】

前記クライアント端末に配信する前記構造化文書に、少なくとも当該構造化文書に一致または含まれる前記操作単位としての部分文書を表す情報が含まれないときは、少なくとも前記操作単位としての部分文書を表す情報と、当該部分文書のデータ形式を他のデータ形式に変換するための少なくとも 1 つの変換規則とを挿入する手段を具備したことを特徴とする請求項 2 4 または 2 5 記載のサーバ装置。

【請求項 2 8】

複数の要素から構成された文書構造を有する構造化文書を表示手段に表示するとともに、この表示された構造化文書に一致または含まれる、少なくとも 1 つの前記要素からなる予め操作単位として定められた部分文書を基に前記表示された構造化文書を編集するためのプログラムであって、

コンピュータに、

前記部分文書には、少なくとも当該部分文書が前記操作単位であることを表す情報が含まれており、この情報を基に、少なくとも前記部分文書と、当該部分文書の前記文書構造上の位置情報とを含む前記部分文書毎の部品データを作成するステップと、

前記構造化文書を前記部品データの集合として記憶手段に記憶するステップと

操作対象として選択された前記部分文書に対する操作内容に応じて、当該操作対象の部分文書に対応する前記部品データを更新することにより前記構造化文書を編集するステップと、

を実行させる文書編集プログラム。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、XML 文書、HTML 文書などの構造化文書の編集方法および装置に関する。

【0 0 0 2】

## 【従来の技術】

WWW (World Wide Web) において、ショッピングサイトでは商品カタログを、ニュースサイトではニュース記事を、一覧列挙するページを持っている。またポータルサイトの多くは、トップページにおいて、ポートレットと呼ばれる部分ページエリアに、株価情報・気象情報・ヘッドラインニュースなど、異なるジャンルの情報を、組み合わせて記載している。ここでは、このジャンル毎の情報をサブコンテンツと呼ぶことにすると、最近は、インターネットを介して閲覧することのできるWebページには、従来の文書的なWebページと異なり、内部に含まれるサブコンテンツ同士の独立性が高く、寄せ集め的に構成されているWebページが増えてきている。

## 【0003】

このようなWebページに対して、ユーザからは、ページ全体を順に読んでいくのではなく、興味のあるサブコンテンツだけを選択的に閲覧したいという要望が高まってきている。閲覧のみに限らず、例えば、以下に示すような要望がある。

## 【0004】

- 1) サブコンテンツ単位で、比較・並べ替えたい
- 2) サブコンテンツ単位で、分類・整理したい
- 3) サブコンテンツ単位で、収集・保存したい

しかし、現在のブラウザでは、上記要望に対応できる機能を持つものはない。

## 【0005】

## 【発明が解決しようとする課題】

このように、従来は、1つのWebページ内全体を閲覧することは可能でも、異なる複数のWebページのそれぞれから、各Webページ内に含まれている、例えば、それぞれが内容的に独立性の高い複数のサブコンテンツのうち、所望のサブコンテンツのみを収集して、サブコンテンツ単位で、比較・並べ替え、分類・整理、保存することはできないという問題点があった。

## 【0006】

そこで、本発明は、上記問題点に鑑み、Webページ単位ではなく、各Web

ページに含まれているサブコンテンツ単位での閲覧を可能にし、しかも、1つまたは異なる複数のWebページのそれぞれから、所望のサブコンテンツのみを収集して、サブコンテンツ単位で編集することのできる文書編集方法およびシステムを提供することを目的とする。

## 【0007】

また、クライアント側で、Webページ単位ではなく、各Webページに含まれているサブコンテンツ単位での閲覧を可能にし、しかも、1つまたは異なる複数のWebページのそれぞれから、所望のサブコンテンツのみを収集して、サブコンテンツ単位で、比較・並べ替え、分類・整理、保存することのできるようなWebページを提供することのできるサーバ装置を提供することを目的とする。

## 【0008】

## 【課題を解決するための手段】

本発明は、複数の要素から構成された文書構造を有する構造化文書を表示手段に表示するとともに、この表示された構造化文書に一致または含まれる、少なくとも1つの前記要素からなる予め操作単位として定められた部分文書を基に前記表示された構造化文書を編集するものであって、前記部分文書には、少なくとも当該部分文書が前記操作単位であることを表す情報が含まれており、この情報を基に、少なくとも前記部分文書と、当該部分文書の前記文書構造上の位置情報とを含む前記部分文書毎の部品データを作成して、前記構造化文書を前記部品データの集合として記憶手段に記憶し、操作対象として選択された前記部分文書に対する操作内容に応じて、当該操作対象の部分文書に対応する前記部品データを更新することにより前記構造化文書を編集することを特徴とする。

## 【0009】

本発明によれば、構造化文書に含まれている、操作単位の部分文書を表す情報を基に識別することのできる各部分文書のそれぞれについて、その付加情報を基に、上記部品データ（ページ部品）を作成することで、このページ部品を基に、画面上に表示された内容をサブコンテンツ単位（部分文書単位）で容易に編集操作を行うことできる。

## 【0010】

従って、構造化文書として、例えばWebページ単位ではなく、各Webページに含まれているサブコンテンツ単位での閲覧を可能にし、しかも、1つまたは異なる複数のWebページのそれぞれから、所望のサブコンテンツのみを収集して、サブコンテンツ単位で編集することができるようになる。

## 【0011】

本発明のサーバ装置は、複数の要素から構成された文書構造を有する構造化文書を表示手段に表示するとともに、この表示された構造化文書に一致または含まれる、少なくとも1つの前記要素からなる予め操作単位として定められた部分文書を基に前記表示された構造化文書を編集するクライアント装置に対し、前記構造化文書を配信するものであって、前記クライアント端末に配信する前記構造化文書には、少なくとも当該構造化文書に一致または含まれる前記操作単位としての部分文書を表す情報が含まれている（すなわち、クライアント端末に配信する構造化文書はブロックWeb文書である）ことを特徴とする。

## 【0012】

本発明によれば、クライアント側で、構造化文書として、例えばWebページ単位ではなく、各Webページに含まれているサブコンテンツ単位での閲覧を可能にし、しかも、1つまたは異なる複数のWebページのそれぞれから、所望のサブコンテンツのみを収集して、サブコンテンツ単位で、比較・並べ替え、分類・整理、保存することができるようなWebページを提供することができる。

## 【0013】

好ましくは、前記クライアント端末に配信する前記構造化文書には、当該構造化文書に一致または含まれる前記操作単位としての部分文書を表す情報とともに、各部分文書について、当該部分文書のタイプと、当該部分文書のデータ形式を他のデータ形式に変換するための少なくとも1つの変換規則と、当該部分文書として前記表示手段に表示される他の構造化文書中の部分文書へのリンク情報と、当該部分文書に挿入可能な他の部分文書のタイプおよび挿入位置のうちの少なくとも1つが、当該部分文書の付加情報として含まれている。

## 【0014】

好ましくは、上記サーバ装置は、前記クライアント端末に配信する前記構造化

文書に、少なくとも当該構造化文書に一致または含まれる前記操作単位としての部分文書を表す情報が含まれないときは、少なくとも前記操作単位としての部分文書を表す情報を挿入する手段を具備する。これにより、ブロックWeb文書でない構造化文書（例えば、標準的なHTML文書やXML文書など）もサーバ装置にて、ブロックWeb文書に変換されるので、クライアント側では、サブコンテンツ単位のWebページの編集が容易に行える。

【0015】

好ましくは、前記クライアント端末に配信する前記構造化文書に、少なくとも当該構造化文書に一致または含まれる前記操作単位としての部分文書を表す情報が含まれないときは、少なくとも前記操作単位としての部分文書を表す情報と、当該部分文書のデータ形式を他のデータ形式に変換するための少なくとも1つの変換規則とを挿入する手段を具備する。

【0016】

【発明の実施の形態】

以下、本発明の一実施形態について図面を参照して説明する。

【0017】

まず、以下の実施形態の説明で用いる用語の説明を行う。

【0018】

Web文書、Webページとは、インターネットを介して閲覧可能な、HTMLやXMLなどで記述された、複数の要素から構成された文書構造を有する構造化文書である。なお、以下の説明では、Web文書（Webページとも云う）を例にとり説明するが、この場合に限らない。インターネットを介して閲覧されることを目的として作成され、インターネットを介して取得した構造化文書でなくとも、任意のエディタで任意の目的で作成され、任意の手段で取得した、HTML文書やXML文書などの一般的な構造化文書であっても、本実施形態は適用可能である。

【0019】

サブコンテンツは、表示されたWebページ中の内容的にまとまりのある、例えば、それぞれが内容的に独立性の高い表示データであって、どのようにサブコ



コンテンツを区切るかは、Webページの作者の意図するところである。サブコンテンツは、Webページの作者によって、全く異なる観点から様々に区切ることができる部分文書に対応し、部分文書で当該表示データを表示することができるようになっている。従って、サブコンテンツは、1つのWebページ(Web文書)中のユーザが独立的に取り扱いたい部分ページ(部分文書)とも言える。例えば、ポータルサイトの多くは、トップページにおいて、ポートレットと呼ばれる部分的なページエリアに、株価情報・気象情報・ニュース記事など、異なるジャンルの情報を、組み合わせて記載している。例えば、ジャンル別の情報、すなわち、例えば、ニュース記事や気象情報のそれぞれが1つのサブコンテンツ(部分文書)であってもよいし、ニュース記事の中の1つ1つのニュースがサブコンテンツ(部分文書)であってもよい。サブコンテンツの内容、区切り方は特に限定するものではない。なお、1つのWebページ全体が1つの部分文書に一致し、部分文書が複数の部分文書を包含してもよい。なお、以下の説明において、サブコンテンツと部分文書とを同等にみなしてもよい。

#### 【0020】

##### (各機能の概要)

本実施形態に係る文書編集システムの概要を説明する。ここでの説明では、広くWeb表示機能を持つソフトウェア一般の総称として、ブラウザと呼んでおり、Internet Explorer(商標)やNetscape(商標)などのWeb表示機能を主眼に置いたソフトウェアに限定しているのではない。たとえば、本実施形態に係る文書編集システムではWeb文書の表示機能を持つデータベースを装備しており、これもブラウザと呼んでいる。

#### 【0021】

- 1) ブラウザ上でのドラッグドロップなどのGUI操作で、サブコンテンツを自由に再配置する機能(比較・並べ替えのニーズに対応)
- 2) あるブラウザから別のブラウザへ、ドラッグドロップなどのGUI操作でサブコンテンツを受け渡す機能(分類・整理のニーズに対応)
- 3) サブコンテンツが含まれていないページを表示するブラウザを開き、さまざまなページから取り込みたいサブコンテンツをドラッグドロップなどのGUI

## I 操作でそのブラウザへ受け渡す機能（収集・保存のニーズに対応）

この基本的な機能仕様により、ユーザからの上記ニーズをカバーすることができる。この基本機能仕様を核として周辺機能を付加し、完全な機能仕様を設計した。

## 【 0 0 2 2 】

図 1 は、それらの機能を俯瞰したものである。以下、図 1 を基に機能仕様について概説する。ただし図 1 は利用形態をわかりやすく説明するための機能フロー例であり、実際のシステム設計での構成は後述のように異なるため、機能ブロック図ではない。

## 【 0 0 2 3 】

本システムのブラウザは、インターネットあるいはファイルシステムなどにある Web ページを読み込むと、従来のブラウザと同様 Web ページを表示するが、それに加えて、サブコンテンツをユーザが便利に取り扱うための多くの操作機能、すなわち、図 1 の 1-1 ~ 1-14 を提供する。以下、これら各機能 1-1 ~ 1-14 について、その概要を説明する。

## 【 0 0 2 4 】

（自由な再配置機能 1-1） 機能の 1 つは、ブラウザ上でのサブコンテンツの自由な再配置機能である。ユーザはブラウザの Web 表示部において、ドラッグドロップなどの GUI 操作を用いて、任意のサブコンテンツを別の位置へ動かすことができる。自由といっても、ドット単位で完全に自由なモードと、枠に沿って配置するレイアウトモードの 2 種類がある。これにより、たとえば、ショッピングサイトでの商品列举ページにおいて、商品の並びをユーザの都合で閲覧中に変更することができるなどの効果がある。

## 【 0 0 2 5 】

（ソート機能 1-2） また、それに関連する機能としてソート機能がある。ソート機能 1-2 は、メニューやツールボタンなどから機能を選ぶことによって起動する。ページに含まれるすべてのサブコンテンツあるいは指定されたサブコンテンツの中で、特定項目の内容にしたがって並べ替えることができる。これにより、たとえば、ショッピングサイトでの商品を値段や種別で並べ替えることが

できるので、商品比較が容易となる効果がある。これに加えて、後述の図1中のページ間受け渡し機能1-4をあわせ用いることで、さまざまなWebサイトやWebページの商品アイテム同士を比較することもできる。

## 【0026】

従来であれば、異なるWebサイトのコンテンツ同士の比較については、直接的な機械の支援を受けることができず、人間が目で比較するしかなかった。

## 【0027】

(検索機能1-3) また、サブコンテンツの検索機能がある。検索機能は、メニューやツールボタンなどから機能を選ぶことによって起動する。ページの中から、ユーザの入力する条件にしたがって検索を行い、その結果を同ブラウザ内で選択状態にする(フォーカスを当てて、ユーザに対してどれが該当しているかを示す)か、図1中の新規空ページ作成機能1-9を用いて別ウィンドウとしてブラウザを開き、そこに該当するサブコンテンツを載せるか、の2種類の方法が提供される。また、この検索機能を繰り返し使うことで、順次検索絞込みをすることもできる。

## 【0028】

(ページ間受け渡し機能1-4) 次の機能として、異なるブラウザで表示されているWebページ同士で、サブコンテンツを受け渡す機能がある。複数のWebサイトあるいはWebページで別々に表示されているサブコンテンツを同じページで見ることができる。この機能には先ほど述べた効果があるが、さらに図1の新規空ページ作成機能1-9と合わせ用いることで、新規のページに興味のあるサブコンテンツだけを収集、さらにその中で比較することができる。

## 【0029】

(形式の調整機能1-5) また、その補助機能として、形式(タイプ)の調整機能がある。異なるWebページ間でサブコンテンツを受け渡す場合、そのサブコンテンツのタイプがそぐわない場合がある。たとえば商品列挙ページはページごとに表示形式やデータ形式が異なる場合がある。そのまま挿入することも可能であるが、ソート1-2などをあわせて利用することでサブコンテンツを比較したい場合には、不便である。形式(タイプ)の調整機能1-5はこの問題を自

動的に解決するもので、システムがサブコンテンツのタイプが受ける側のページに合わないとは判断すると、システムが持つWeb形式変換ルールのデータベースから適切なものを探し出し、変換ルールを適用してタイプをそろえてから挿入する。

#### 【 0 0 3 0 】

(デザイン切替 1 - 6) その他の機能としてデザイン (表示形式あるいはスタイルとも云う) の切り替えがある。デザインの切り替え機能はメニューやツールボタンなどから機能を選ぶことによって起動する。デザインの切り替え機能は、単一あるいは複数のサブコンテンツに対して、コンテンツはそのままにデザインを切り替える方法である。たとえば、Webに列挙されている商品カタログに対して、データスペックがわかりやすい表示形式や商品デザインがわかりやすい表示形式に切り替えることができる。ただしこの機能は、WebページがXMLなどのようにデータとスタイルシート (表示方法) を分離して記述されている場合にのみ利用可能である。

#### 【 0 0 3 1 】

(挿入機能 1 - 7) サブコンテンツを他のサブコンテンツ内に子挿入する機能がある。この機能を利用することで、たとえばユーザは商品カタログの中に商品カタログを収めることができる。この機能は再帰的であり、サブコンテンツの中にサブコンテンツを入れ、また内部のサブコンテンツに別のサブコンテンツを入れるという処理を無限に行うことができる。

#### 【 0 0 3 2 】

(受け入れ判定機能 1 - 8) サブコンテンツの受け入れ判定機能 1 - 8 は、上記した自由な再配置機能 1 - 1、ページ間受け渡し機能 1 - 4 など、ドラッグドロップなどのGUI操作で別の場所にサブコンテンツを挿入する際に自動的に起動される機能である。この機能により、挿入動作を実行する際、受け入れる側となるWebページ、あるいは挿入機能 1 - 7 を利用した場合は、別のサブコンテンツが挿入を受け入れるための条件が設定されていた場合、その条件を満たさない場合は、挿入処理をキャンセルする。受け入れ条件としては、サブコンテンツの形式に関して与えることができる。

## 【 0 0 3 3 】

この機能 1 - 8 は形式の調整機能 1 - 5 と連動する。もし挿入対象のサブコンテンツが受け入れ側の条件を満たさない場合、形式の調整機能 1 - 5 の機能を用いて、受け入れ可能なタイプに変換できるかどうかを試みる。もし受け入れ可能なタイプへの変換方法が見つかった場合にはそのタイプで挿入される。もし見つからなければ、挿入処理はキャンセルされる。

## 【 0 0 3 4 】

(新規空ページ作成機能 1 - 9) 新規の空ページを作成する機能 1 - 9 は、メニューやツールボタンなどから機能を選ぶことによって起動する。実行すると、別のブラウザウィンドウを開き、そこに空の Web ページを新規作成する。ひとつの重要な活用方法として、このページをスクラップブックとして、複数の Web ページからユーザが興味のあるサブコンテンツのみをドラッグドロップなどで収集することができる。この新規ページは、開く元となったブラウザの Web ページのデザインを継承することも可能であり、その場合、元の Web ページのうちサブコンテンツのみをすべて取り除いた状態で、別のウィンドウに新規ページとして表示される。

## 【 0 0 3 5 】

(コンテンツ編集機能 1 - 1 0) コンテンツ編集機能 1 - 1 0 は、メニューやツールボタンなどから機能を選ぶことによって起動する。起動されると編集モードとなり、ユーザは、ブラウザ上であるいは別ウィンドウの専用編集ソフト上で、現在選択しているサブコンテンツを編集することができる。

## 【 0 0 3 6 】

編集機能としては、表示されているデザインのまま、HTML フォーム形式で各データフィールドを変更するインラインモード、またそれだけでなくデザインを変更する完全モードがある。またこれらのモードに対する付加機能として、XML のようにデザインとデータが分離されている場合、データフィールドを変更した場合に、内部のデータ部に値を反映する (書き戻す) 機能、データ形式がスキーマなどで厳密に指定されている場合に、そのデータ形式に沿って最適な入力 HTML フォーム GUI を提供する機能などがある。これらの機能は、既存の技

術を用いれば、実現可能である。

【 0 0 3 7 】

また、サブコンテンツを新規作成することもできる。この機能は、インラインモードの機能を利用しており、起動すると、直前のサブコンテンツの形式でHTMLフォームが表示される。ユーザはフォームの各項目に値を埋めて確定させると、新しいサブコンテンツがそのページ内に追加される。

【 0 0 3 8 】

(保存機能 1 - 1 1) 保存機能 1 - 1 1 は、メニューやツールボタンなどから機能を選ぶことによって起動する。ユーザは今までに紹介した機能を用いることにより、Web ページの構成やコンテンツを編集することができる。保存機能 1 - 1 1 を用いることで、変更された状態をさまざまな記憶システムに保存することができる。保存先としてはユーザが使用しているローカルホストのファイルシステムだけでなく、自分のホームページなどが置いてあるWeb サーバ、データベースなどを選択することができる。

【 0 0 3 9 】

保存の形式としては、HTML や XML に加えて、これらサブコンテンツの構成情報を付加した本システムの特殊形式を選択することができる。特殊形式は、どれがサブコンテンツなのかをあわせて保存するので、次回に読み込んだときにも、前述したサブコンテンツ向けの機能を引き続き利用することができる。

【 0 0 4 0 】

(データソース更新機能 1 - 1 2) その補助的機能としてデータソース更新機能 1 - 1 2 がある。この機能は、保存機能の 1 種であるが、もともとWeb サーバから送られてきたWeb ページを編集した場合、その更新情報を元のWeb サーバに送信することができる。受信したWeb サーバは、そのシステム構成に応じてデータベースなどの適切なストレージにその更新データを反映させる。

【 0 0 4 1 】

(外部アプリケーションへの取り込み機能 1 - 1 3) 外部アプリケーションへのサブコンテンツの取り込み機能 1 - 1 3 は、ブラウザで開いている任意のWeb ページ (スクラップブックを含む) から気に入ったサブコンテンツだけをド

ラッグドロップなどのGUI操作でアプリケーションに渡すことで、そのアプリケーションに取り込む機能である。取り込み先としては、メール・OAツール・社内ワークフローシステムなどが有効なアプリケーションの一例として挙げられる。

#### 【0042】

(付箋紙としてデスクトップへの貼り付け機能1-14) 上記外部アプリケーションへの取り込み機能1-13に準ずる機能として、デスクトップへサブコンテンツの貼り付け機能1-14がある。ブラウザで開いている任意のWebページ(スクラップブックを含む)から気に入ったサブコンテンツをデスクトップにドロップすることで、付箋紙(電子付箋)として貼り付けておくことができる。この付箋紙に貼り付けたサブコンテンツは、後から任意のページやアプリケーションへドロップすることもできる。

#### 【0043】

(システム構成)

上記各機能を実現するための文書編集システムの構成について説明する。

#### 【0044】

サブコンテンツを処理単位として上記各機能をユーザに提供するには、少なくとも、Webページの中で、どこがサブコンテンツに当たるのか(サブコンテンツが他のサブコンテンツに包含される(親のサブコンテンツをもつ)こともあれば、他のサブコンテンツを包含する(子のサブコンテンツをもつ)こともある(これをサブコンテンツ(部分文書)間の包含関係あるいは親子関係と呼ぶ)。そこで、このサブコンテンツ間(部分文書間)の包含関係に関する情報が必要となる。これをWebブロック構造情報あるいは簡単にブロック構造情報と呼ぶ。また、1つのWebページをその中に含まれているブロック構造情報を基に、本システムの処理単位であるサブコンテンツ単位に分解することをブロック化とも云う。

#### 【0045】

また、ブロック化というのは本発明に特徴付けるものであって、現在広く用いられているHTML文書やXML文書などのWeb文書には、もちろんブロック

構造情報は含まれていない。

【0046】

次に、このブロック構造情報をどのようにして得るのか、そしてどう取り扱うか、について説明する。ここでは、ブロック構造情報の取得および取扱に際し、次のような方針を採用した。

【0047】

1) ブロック構造情報も含めて記述できる専用Web言語を準備する（以降、この言語をブロックWeb言語、この言語で書かれた文書をブロックWeb文書と呼ぶ）。

【0048】

2) 本システムはブロックWeb言語を元に処理を行うようにする。

【0049】

3) 本システムにHTML、XMLなど従来からあるWeb文書が入力された場合は、上記処理を行う前にブロックWeb言語へ変換してから取り込む。この際、ブロック構造情報を補う必要があり、補う方法としていくつかの機能を提供する。

【0050】

上記方針を採った理由は2つである。

【0051】

まず、完全な情報を持つこの専用のブロックWeb言語を、本システムを活用するための主要な方法とする意図がある。ブロックWeb言語を入力とする場合、作者が、自分の演出したい意図に基づいて、コンテンツとともにどこをブロック化（独立的に取り扱うサブコンテンツ）したいのかを記述することができる。

【0052】

したがって、Web作者に使用してもらうために、このブロックWeb言語は広く公開することを想定する。

【0053】

なお、ブロックWeb言語を基に、ユーザがサブコンテンツを取り扱うための操作機能を提供する方式として、特開2001-109742号公報記載の技術



を利用する。

【 0 0 5 4 】

その一方で、一般的に普及している Web ページ記述言語に柔軟に対応することを目的とする。現時点で普及している Web 言語は HTML および XML である。これらを直接扱うための処理モジュールを用意するのではなく、先ほどのブロック Web 言語へ変換する方式を取る。こうすることにより、システムがシンプルになると同時に、変換処理だけを独立させることができるので、今後必要となる変換方式に対してもプラグインとして対応するといった拡張性を得るメリットがある。

【 0 0 5 5 】

変換といっても、HTML や XML 自体はブロック構造情報を持たないので、不足している情報を補わなければならない。これに対して、あらかじめ別途用意した別の文書から与える方法、ユーザがブラウザ使用時にその場その場で指定する方法、文書内容からブロック構造を機械的に推測するいわゆるテキストマイニング的方法がある。なお、以下の説明では、前 2 つの方法を用いて実現する場合を示す。3 つ目の方法も既存の技術を用いることで実現可能だが、ここで可能性を示すにとどめる。

【 0 0 5 6 】

以上を踏まえて構成された文書編集システムの機能的な構成例を図 2 に示す。また図 4 は本システムが扱うグローバルなデータオブジェクトの構成例を示したものである。まず、本システムが処理対象とするオブジェクトデータとしてのページ部品から説明する。

【 0 0 5 7 】

本システムでは、図 4 に示したような構成の、ページ部品単位のデータオブジェクトを取り扱う。ページ部品とは、各サブコンテンツ（部分文書）に 1 対 1 に対応するデータであり、サブコンテンツの内容自体はもちろんのこと、本システムが提供するサブコンテンツ向けの機能に必要な管理情報を併せ持っている。

【 0 0 5 8 】

図 4 に示したページ部品（データオブジェクト）D 1 は、複数のサブデータオ

プロジェクトD11～D18から構成されている。図4において、例えば、受理情報D14は、さらに、その内部に挿入場所情報D141、受理タイプ情報D142を含むことが表されている。同様に、子ページ部品管理情報D16は、その内部に挿入場所情報D161、子ページ部品への参照情報D162を含む。また、例えば、受理情報D14や子ページ部品管理情報D16のように、複数重なって（ここでは、例えば2重に重なって）表記されたサブデータオブジェクトは、そのようなサブデータオブジェクトが複数存在することを示している。

## 【0059】

ページ部品D1には、タイプ名D11とコンテンツD12とスタイルシートD13と受理情報D14と親ページ部品への参照D15と子ページ部品管理情報D16とリンクURLD17とリンクフラグD18という情報が含まれている。これらの情報は、ブロックWeb文書から抽出される各部分文書を構成する要素として含まれていたり、属性として含まれている情報などである。

## 【0060】

タイプとは、部分文書のデータ形式（形式的な違い）や、当該部分文書の内容に関するカテゴリ・ジャンルの違い（意味的な違い）に応じて各部分文書を区別するためのもので、このような部分文書の意味的、形式的な違いに応じた各タイプを区別するための識別情報がタイプ名である。ページ部品D1に含まれるタイプ名D11には、当該ページ部品D1に対応する部分文書に対応するタイプ名が保持されている。このタイプ名は、部分文書のトップノードであるエレメント（要素）に付される「type」属性で記述されている。

## 【0061】

コンテンツD12には、当該ページ部品D1に対応する部分文書が保持される。表示データが、当該ページ部品を含む構造化文書以外の他の構造化文書中のものであるときには、当該表示データへのリンク情報がリンクURLD17に格納される（また、このとき、リンクフラグD18がセットされる）ので、この場合は、コンテンツD12は空でもよい。

## 【0062】

スタイルシートD13は、コンテンツD12に保持されている部分文書に適用

可能なデータ形式の変換規則（スタイルシート）や、当該部分文書に適用可能な表示形式別に、当該表示データのデータ形式を当該表示形式対応のデータ形式へ変換するための変換規則（スタイルシート）などが保持されている。当該表示データに複数の変換規則が適用可能であれば、複数のスタイルシートD13がページ部品D1に含まれていることになる。スタイルシートは、部分文書のトップノードであるエレメント（要素）に記述される「style」属性で記述されている。

## 【0063】

受理情報D14は、当該ページ部品D1に対応する部分文書が受け入れ（挿入）可能な部分文書のタイプ名を受理タイプD142として保持し、文書構造上の挿入場所D141とを保持する。この受理タイプD142は、部分文書中の「accept」エレメント（の属性）として記述されており、挿入場所D141は、この「accept」エレメントの文書構造上の配置位置に対応する位置情報である。

## 【0064】

親ページ部品への参照D15は、当該ページ部品D1に対応する部分文書が他の部分文書に包含されている場合、当該ページ部品D1と親である他の部分文書に対応するページ部品とを関連付けるための情報である。

## 【0065】

子ページ部品管理情報D16は、当該ページ部品D1に対応する部分文書が他の部分文書を包含している場合、当該ページ部品D1と子供である他の部分文書に対応するページ部品とを関連付けるための情報であり、文書構造上の当該他の部分文書の挿入場所D161と、当該他の部分文書の文書構造上の位置情報としての子ページ部品への参照D162とを保持する。挿入場所D161は、部分文書中の、他の部分文書を挿入することを指示するエレメント（例えば、後述する「import」エレメントなど）の配置位置であり、このエレメントの例えば属性として、挿入する部分文書の位置情報としての子ページ部品への参照D162が記述されている。

## 【0066】

なお、図4に示した各サブデータオブジェクトについては、以下に示す、図2に示したシステムの構成や動作の説明の中で必要に応じて説明する。

## 【0067】

また、サブコンテンツとは、1つのWebページ（Web文書）中のユーザが独立的に取り扱いたい部分ページ（部分文書）に対応すると説明してきた。本システムでは、サブコンテンツに1対1に対応するオブジェクトとして、図4に示したようなページ部品D1を取り扱うが、ページ全体（すなわち、1つのWebページ）も1つのページ部品として処理する。

## 【0068】

ページ部品について、図3を用いて具体的に説明する。図3には、1つのWebページとしてのページ部品3-1について示している。ページ全体に対応するページ部品3-1には、複数のページ部品3-2～3-7が含まれており、本システムでは、ページ部品3-1と、その中に含まれるページ部品3-2～3-7とは、親子関係にあるものとして管理する。ページ全体もページ部品とみなすことで、特別な処理が不要になり、システムがシンプルになると同時に、ページ全体を他のページの一部として取り込むニーズがあってもよいと考えるからである。

## 【0069】

次に、図2に示した文書編集システムの各機能モジュールについて説明する。

## 【0070】

図2に示すように、文書編集システムは、大きく分けて、ホストモジュールM1と、文書処理モジュールM2と、デスクトップ貼り付けモジュールM3とから構成されている。これらモジュールM1～M3は、それぞれ独立したモジュールであり、ホストモジュールM1は、文書処理モジュールM2を必要に応じて実体化して呼び出し、文書処理モジュールM2は、デスクトップ貼り付けモジュールM3を必要に応じて実体化して呼び出すようになっている。

## 【0071】

ホストモジュールM1は、文書処理モジュールM2を受け入れる（ホストする）コンテナコンポーネントである。基本的な機能は、OS（Operating

S y s t e m) 上にウィンドウフレームを構築し、文書処理モジュールM2が機能するためのウィンドウ領域を確保すること、またユーザが操作するためのメニューやツールボタンなどを提供し、入力されたコマンドを文書処理モジュールM2に伝えることである。

【0072】

ホストモジュールM1としては、具体的には、W e b ブラウザとページ部品アルバムホスト（図5に示すようなページ部品管理に特化した機能を提供する専用アプリケーション）を想定している。I n t e r n e t E x p l o e r（商標）やN e t s c a p e（商標）などの一般的なW e b ブラウザは、汎用文書サービスホストとして機能するよう作成されているので、それぞれのブラウザの仕様にしたがって、文書処理モジュールM2を、当該ブラウザから呼び出して起動可能なように、文書処理モジュールM2に関して必要な登録処理を行えばよい。この詳細はそれぞれ汎用ブラウザのソフトウェア仕様の範疇であり、これ以上の説明は省略する。なお、ここでは、上記一般的なW e b ブラウザをホストモジュールとして用いた場合、このホストモジュールを第1のホストモジュールM1と呼ぶ。

【0073】

一方、本実施形態に係る文書編集システム専用のホストモジュールを第2のホストモジュールM1と呼び、これについて、後述する。

【0074】

文書処理モジュールM2は、ブロック化機能およびサブコンテンツ向けの機能を持つW e b 文書処理モジュールであり、ホストモジュールM1からW e b ページの数だけ実体化され呼び出される。すなわちW e b ページ1つに対して、文書処理モジュールが1つメモリ上に生成される。

【0075】

文書処理モジュールM2は、大きく分けてページ読込モジュールM21とブロックW e b 文書処理モジュールM22という2つのサブモジュールを含む。

【0076】

ブロック化機能は、ページ読込モジュールM21が担当する。ページ読み込み

モジュールM21は、URL（あるいはローカルマシン上であればファイルパス）で指定された文書を読み込んだ上で、もし、ブロックWeb文書ではないHTML文書やXML文書などの一般的なWeb文書であると判断された場合、複数の方法のうちの1つを実行して、ブロックWeb文書に変換する。

## 【0077】

サブコンテンツ向けの機能を提供するのは、ブロックWeb文書処理モジュールM22である。モジュールM22は、ページ部品管理モジュールM221、レンダリング処理モジュールM222、GUI処理モジュールM223、エクスポートモジュールM224、ページ部品形式調整モジュールM225という5つのサブモジュールを含む。

## 【0078】

ページ部品管理モジュールM221は、各ページから抽出されたページ部品を保持するページ部品データベースM221bを持ち、ページ部品（サブコンテンツ）単位に挿入や移動や削除などページの構成を変更する機能や、これらを保存する機能を持つ。

## 【0079】

レンダリング処理モジュールM222は、ページ部品管理モジュールM221で管理されているページ部品群をWebページとして表示する機能を持つ。

## 【0080】

GUI処理モジュールM223は、レンダリング処理モジュールM222で表示されたWebページ上で、ドラッグドロップやコンテキストメニューなどサブコンテンツを操作するためのGUI機能を提供する。GUI操作が行われたら、ページ部品管理モジュールM223に対して、当該操作に対応するデータ処理機能呼び出した上で、レンダリング処理モジュールM222を用いて、表示画面を書き換える。

## 【0081】

GUI処理モジュールM223は、同時にエクスポートモジュールM224を呼び出す。エクスポートモジュールM224は、外部のアプリケーション（外部アプリ）に対して、ページ部品を受け渡すための処理機能を提供するものである。

## 【0082】

具体的には、GUI操作においてドラッグドロップやコンテキストメニュー（表示されている任意のGUI部品上で例えば右クリックすると、この部品に関して表示されるもの）でコピー命令などが実行された時、GUI処理モジュールM223はエクスポートモジュールM224を呼び出し、他のアプリケーションやプロセス（これらをまとめて外部アプリと呼ぶ）が読み込むことのできる共有メモリ領域（クリップボード）に対して、外部アプリが読み込める形式で当該ページ部品の内容を書き込む。これにより、外部アプリに対してドロップ操作が行われた場合、外部アプリは共有メモリ領域から内容を読み込むので、ページ部品の受け渡しを行うことができる。

## 【0083】

ページ部品形式調整モジュールM225は、文書処理モジュールM2内あるいは文書処理モジュール間でページ部品の受け渡しを行う際に、ページ部品のサブコンテンツのデータ形式などの調整を行うモジュールであり、ページ部品管理モジュールM221が、ページ部品の挿入操作を行う際に呼び出される。具体的には挿入操作を行う際に、新しく追加するページ部品のタイプ名が、挿入元のページ部品に含まれる受理タイプD142（これを受け入れ可能なタイプ名リストとも呼ぶ）にない場合に、ページ部品形式調整モジュールM225は、変換ルールデータベースM225b（図17参照）から適切な変換ルールの組み合わせを探し出し、ページ部品を受け入れ可能なページ部品に変換して、ページ部品管理モジュールM221に返す。もし解決できる変換ルールの組み合わせが見つからなかった場合、その旨をページ部品管理モジュールM221に通知し、ページ部品管理モジュールは挿入動作をキャンセルする。

## 【0084】

デスクトップ貼り付けモジュールM3は、アプリケーションが持つ文書の一部を、ドラッグドロップ操作によって付箋紙としてデスクトップに貼り付ける機能をユーザに提供する。

## 【0085】

図 2 4 は、図 2 に示した文書編集システムを例えばコンピュータ（計算機）上で実現する場合の構成例を示したものである。

【 0 0 8 6 】

すなわち、図 2 4 において、文書編集システムは、プロセッサ（CPU）2 0、メモリ 2 2、ディスプレイやプリンタなどの出力装置 2 4、マウスやキーボードなどの入力装置 2 3、記憶装置 2 1 から構成されている。

【 0 0 8 7 】

これらのうち、記憶装置 2 1 は、システムの制御の中枢を担うプログラム等の格納やデータ等の一時保持などに利用されるものであって、ホストプログラム 3 1、文書処理プログラム 3 2、デスクトップ貼り付けプログラム 3 3、各種アプリケーションプログラム 3 4 などを格納している。また、メモリ 2 2 は、上記各プログラム実行の際に利用可能な共有メモリやプログラム実行の際のワーキングエリアなどに利用されるものである。また、プロセッサ 2 0 は記憶装置 2 1 内のプログラムを実行することにより、入出力制御や各種処理を含め、必要な各種制御処理を実施するものである。

【 0 0 8 8 】

プロセッサ 2 0 は、主に、記憶装置 2 1 内のホストプログラム 3 1 を実行することにより、図 2 4 に示した文書編集システム上で、図 2 に示した文書編集システムのホストモジュール M 1 に対応する機能を提供するようになっている。

【 0 0 8 9 】

プロセッサ 2 0 は、主に、記憶装置 2 1 内の文書処理プログラム 3 2 を実行することにより、図 2 4 に示した文書編集システム上で、図 2 に示した文書編集システムの文書処理モジュール M 2 に対応する機能を提供するようになっている。

【 0 0 9 0 】

プロセッサ 2 0 は、主に、記憶装置 2 1 内のデスクトップ貼り付けプログラム 3 3 を実行することにより、図 2 4 に示した文書編集システム上で、図 2 に示した文書編集システムのデスクトップ貼り付けモジュール M 3 に対応する機能を提供するようになっている。

【 0 0 9 1 】



さらに、プロセッサ20は、主に、記憶装置21内の各種アプリケーションプログラム34を実行することにより、図24に示した文書編集システム上で、これら各アプリケーション対応の機能を提供するようになっている。

【0092】

以下、図2に示した、各モジュールについて詳細に説明する。

【0093】

(ホストモジュール)

まず、ホストモジュールM1について説明する。前述したように、Internet ExplorerやNetscapeなどの一般的な汎用ブラウザをホストモジュールとして利用する方法と、専用のホストモジュールを提供する方法とがある。

【0094】

一般的な汎用ブラウザをホストモジュールM1として使用する場合（この場合のホストモジュールM1を第1のホストモジュールM1と呼ぶ）は、それぞれのブラウザの仕様にしたがって、文書処理モジュールM2に関して必要な登録処理を行えばよい。この詳細はそれぞれ汎用ブラウザのソフトウェア仕様の範疇であり、これ以上の説明は省略する。

【0095】

ここでは、本実施形態に係る文書処理システム専用のホストモジュールである第2のホストモジュールM1について説明する。

【0096】

第2のホストモジュールM1は、ページ部品を含むページ、すなわち、ブロックWEB文書を、内部に複数保持することができ、これら複数のページをツリーで選択・管理することができるソフトウェアアプリケーションである。実行画面を図6に示す。

【0097】

第2のホストモジュールM1が起動されたときに、ディスプレイに表示される画面上の表示領域は、分類ツリー表示部6-1とWeb表示部6-2から構成されている。

## 【0098】

分類ツリー部6-1の各ノードには、第2のホストモジュールM1の保持する複数のWebページがそれぞれ割り当てられており、選択されているノードに対応するWebページがWeb表示部6-2に表示されている。なお、図6において、Web表示部6-2で表示されているWebページは図3で示したページ部品と同じものである。

## 【0099】

ユーザは分類ツリー部6-1に表示されているノードのうち、所望のノードをクリックすることで、ページ部品としての複数のWebページを閲覧することができる。またユーザ操作により、ツリー上に、ノードを新規追加したり削除したり移動したりすることができる。これらの操作は広く使われているExplorer（商標）と同じ使い勝手である。

## 【0100】

またユーザは、現在、Web表示部6-2に表示されているページ部品をドラッグし、任意のノードへドロップ、あるいは別途開いている第2のホストモジュールM1のWeb表示部6-2にドロップすることで、ページ部品を移動したりコピーしたりすることができる。

## 【0101】

また、Web表示部6-2でさまざまなGUI操作を行うことにより、その他のさまざまなページ部品向けの機能を実行することができる。これらの機能は、後ほど詳細に説明する文書管理モジュールM2によって提供されるものであり、ここでは、その詳細は触れない。

## 【0102】

なお、もう1つのホストモジュール例として前述した一般的なブラウザを用いて構成された第1のホストモジュールM1の場合は、当該第1のホストモジュールM1を起動した際に表示される画面上には、Web表示部6-2のみが存在し、同様に、Web表示部6-2でさまざまなGUI操作を行うことにより、さまざまなページ部品向けの機能を実行することができる。

## 【0103】

第2のホストモジュールM1の機能的な構成について、図5を参照して説明する。

【0104】

第2のホストモジュールM1は、大きく分けて、上記分類ツリー部6-1におけるWebページのツリーについての表示や操作をなどを行うための分類ツリーモジュールM11と、ファイルへの書き出しや読み込みを担当する保存機能モジュールM12と、リジューム機能モジュールM13とから構成される。

【0105】

なお、Web表示部6-2にWebページを表示する機能は、後述する文書処理モジュールM2が担当するようになっている。

【0106】

分類ツリーモジュールM11は、分類ツリーを表示するとともに、表示した分類ツリーに対し、ノードの追加・削除・移動と言ったGUI操作を行うためのものである。上記のツリー表示・GUI操作行うためのプログラム部品として、一般に「ツリーコントロール」と呼ばれるプログラムGUI部品がさまざまなベンダから提供されており、それを使用することで対応することができる。

【0107】

なお、ツリーコントロールとは、開閉可能なツリー上に名前付きノードを表示し、かつ、各ノードをクリックするなどのユーザの操作を通じて、任意の付加機能を割り当てることのできるGUI部品である。

【0108】

分類ツリーモジュールM11は、ツリーコントロールをベースに、それに、第2のホストモジュール独自の機能を追加することで構成するようにしてもよい。なお、ツリーコントロール自体については、本発明の要旨ではないので、その説明は省略する。

【0109】

第2のホストモジュールM1では、ツリーの各ノードそれぞれに対してWebページを割り当てるようになっている。そこで、ツリーコントロールの各ノードには、当該ノードに対応する文書処理モジュールM2を関連付けて当該モジュール

ルM2を参照可能にするために、当該文書処理モジュールM2への参照情報を持つようにしている。

【0110】

ノード選択モジュールM11は、ツリーコントロールでノードのいずれかがクリックされたときに呼び出されるプログラムモジュールであり、一般には選択イベントハンドラーとも呼ばれる。

【0111】

ノード選択モジュールM111は、選択されたノードが当該ノードに関連付けられている文書処理モジュールM2に対して、当該ノードに対応するWebページを表示するように指示を出す。これにより、ノードが選択されるたび、そのノードに関連付けられている文書処理モジュールM2により、該当するWebページが表示される。

【0112】

ノード追加モジュールM112は、ツリーコントロールでノードのいずれかが追加されたときに呼び出されるプログラムモジュールであり、一般には追加イベントハンドラーとも呼ばれる。ノード追加モジュールM112は、文書処理モジュールM2を新規生成し（なおこの状態ではWebコンテンツの中身は空っぽである）、新規に追加されたノードに対して、当該文書処理モジュールM2を関連付けるための参照情報を付加する。

【0113】

ノード削除モジュールM113は、ツリーコントロールでノードのいずれかが削除されるときに呼び出されるプログラムモジュールであり、一般には削除イベントハンドラーとも呼ばれる。ノード削除モジュールM113は、削除されたノードが有する参照情報により、当該ノードに対応付けられている文書管理モジュールM2を終了させ、メモリから消去する。

【0114】

ノード移動モジュールM114は、ツリーコントロールでノードのいずれかが移動させるときに呼び出されるプログラムモジュールであり、一般には移動イベントハンドラーとも呼ばれる。移動イベントモジュールM114は、特に処理を

行わない。

#### 【0115】

ページ部品受け取りモジュールM115は、ツリーコントロールでノードのいずれかに対して、ページ部品がドロップされたとき（移動先を指示されたとき）に呼び出されるプログラムモジュールであり、一般にはドロップイベントハンドラーとも呼ばれる。ページ部品受け取りモジュールM115は、当該ノードに参照情報にて関連付けられている文書管理モジュールM2に対して、受け取ったページ部品データ（シリアルライズイメージ）を基にページ部品を追加するよう命令する。これにより、ドラッグされたページ部品と同じページ部品がドロップ先のノードに対応するWebページに作成される。

#### 【0116】

以上が分類ツリーモジュールM11の説明である。

#### 【0117】

保存機能モジュールM12は、アプリケーションが終了するときに実行されるプログラムモジュールであり、終了イベントハンドラーとも呼ばれる。保存機能モジュールM12は、各ノードが対応付けられているWebページに関する情報をファイルへ書き出す機能を持っている。これを実現するため、ツリーの各ノードを順にたどりながら、そのノードに関連付けられている文書処理モジュールM2に対して保存命令を出す。

#### 【0118】

保存の方法として、それぞれの文書を別々に保存することがまず考えられる。しかし、それぞれのファイル名が重複しない工夫が必要な上、次回に読み込む際にもファイルを拾い集めなければならない、手間がかかる。そこで、すべての文書を1つのファイルにまとめて保存する方法も有効である。

#### 【0119】

リジューム機能モジュールM13は、アプリケーションが起動されたときに実行されるプログラムモジュールであり、初期化イベントハンドラーとも呼ばれる。リジューム機能モジュールM13は、前回終了したときのWebページなどに関する情報をメモリに戻し、アプリケーションを前回と同じ復元する機能を持つ

ている。これを実現するため、保存されたファイルを順に開いていき、ツリー上のノードおよびそのノードに関連付ける文書処理モジュールM2を順に再生成する。

# 【0120】

(文書処理モジュール)

次に、文書処理モジュールM2について説明する。文書処理モジュールM2を構成する各サブモジュールの詳細説明をするのに先立ち、当該モジュールM2が取り扱うブロックWeb文書について説明する。

# 【0121】

以下に、ブロックWeb文書の一例を示す。

# 【0122】

```
<?xml version="1.0"?>
<root style="root.xsl">
  <a component="yes" type="type-a" style="1.xsl;a2.xsl;a3.xsl">
    <a>aaaaa</a>
    <a>aaaaaaaa</a>
  </a>
  <b component="yes" type="type-b" style="b1.xsl;b2.xsl;b3.xsl">
    <b>bbbbbb</b>
  </b>
  <accept type="type-a;type-b"/>
  <import href="http://www.shiba.co.jp/aaaaa.pzx#xpointer(/root/c)"/>
</root>
```

上記のブロックWeb文書は、「root(ルート)」エレメント以下に配置されている、「component」属性が含まれるエレメントにて、独立的に取り扱うことのできるページ部品に対応する部分文書「a」および「b」が含ま

れていることを示している。部分文書「a」で表示される表示データは、「a a a a a」や「a a a a a a a」であり、部分文書「b」で表示される表示データは、「b b b b b」である。さらに、rootエレメントから始まる当該Web文書には、「accept」タグで示される位置に、「type-a」タイプあるいは「type-b」タイプのページ部品を挿入できることが示されている。

## 【0123】

なお、上記のブロックWeb文書から明らかなように、サブコンテンツに対応する構造化文書中の部分文書は、当該部分文書中のトップノード（あるいはルートノードである場合もある）に記述されている「component」属性にて、その値が「yes」であることにより判別することができる。ここでは、「component」属性の記述されているノードをトップノードとする部分文書が1つのサブコンテンツに対応すると判別している。

## 【0124】

このブロックWeb文書は、特開2001-109742号公報で開示されているWeb文書を拡張したものである。

## 【0125】

また、上記ブロックWeb文書において、「import」というタグは特殊な意味を持ち、href属性で指定されたURL（上記例ではXPointerつきURLを用いてページの一部を指定）のページ部品が当該位置に挿入されるという機能を持っている。外部のリソースとしてページ部品であるということを示すものであって、その他、ページ内での取り扱いは他のページ部品とまったく同じである。

## 【0126】

また、HTTP（Hypertext Transfer Protocol）を介して渡される場合は、「Content-Type」は、「text/cmp」でなければならない、FTP（File Transfer Protocol）などその他のネットワークプロトコルやファイルシステムからの入力であれば、拡張子が「cmp」でなければならないというようにルールを決めている。これは入力文書がブロックWeb文書であるかどうかを、文書編集システムが

判定するために用意するルールである。

【0127】

(ページ読込モジュール)

図7は、ページ読込モジュールM2-1の構成を示したものであり、矢印はデータの流れを示している。

【0128】

ページ読込モジュールM21では、主にWebページのブロック化を行うようになっている。文書処理モジュールM2への文書の入力を管理しており、与えられた文書がブロックWeb文書であれば、それはそのままページ読込モジュールM21を通過させて、ブロックWeb文書処理モジュールM22に渡すが、それ以外のWeb文書（たとえばHTML文書やXML文書など）であれば、ブロックWeb文書に変換して、ブロックWeb文書処理モジュールM22に渡す。

【0129】

ページ読込モジュールM21はページタイプ判定モジュールM211、ブロック化モジュールM212から構成される。ページ読込モジュールM21に入力されるものは、HTTPなどの通信あるいはファイルシステムなどから投入されたWeb文書である。HTTP通信機能やファイルシステムからの読み込み処理については、周知の技術を用いればよく、ここでの説明は省略する。

【0130】

入力文書はまずページタイプ判定モジュールM211に渡される。ページタイプ判定モジュールM211は、図8に示す手順で、入力文書がブロックWeb文書かどうかを判定するようになっている。すなわち、入力文書がHTTPを介して渡される場合は、「Content-Type」が、「text/cmp」であれば、当該入力文書はブロックWeb文書であり（ステップS1～ステップS3）、FTP（File Transfer Protocol）などその他のネットワークプロトコルやファイルシステムからの入力であれば、拡張子が「cmp」であれば、当該入力文書はブロックWeb文書であると判定する（ステップS1）。

【0131】



図7に戻り、ページタイプ判定モジュールM211により、ブロックWeb文書であると判断されれば、その文書をそのままページ読込モジュールM21の出力とする。ブロックWeb文書でないと判断されれば、ブロック化モジュールM212によって、ブロックWeb文書へ変換する処理を行い、その結果を出力とする。

#### 【0132】

ブロック化モジュールM212は、与えられた文書に対し、ブロック化するための変換ルール（ここでは、ブロックシートと呼ぶ）を用いて変換処理を実行し、その結果得られたブロックWeb文書を出力する。変換ルールを蓄積しているのがブロックシートデータベース（DB）M214であり、入力文書のURLをキーとして、当該入力文書に対応する変換規則を検索可能なようになっている。入力文書が、HTTPでなくFTPやファイルシステムからの入力である場合は、FTPのURL、ファイルURLが用いられる。変換ルールとして、本システムではXSLTを用いる。

#### 【0133】

ブロックシートDBM214を用いて、実際に変換処理を行うのは、ブロックシート適用モジュールM213である。この処理の流れを図9に示す。この出力はブロックWeb文書処理モジュールM22に渡される。

#### 【0134】

ここで、入力文書に対応するブロックシートが見つからなかった場合（ステップS12）の処理について補足する。入力文書に対応するブロックシートがない場合、ブロック構造情報を持たないため、ページ全体が1つのページ部品と解釈される。すなわち、当該入力文書のルートノードに「component」属性を付与する。また、ページ部品では本来指定されるはずのスタイル指定（style属性）が存在しないので、本システムが内部に持っているデフォルトスタイルを与える（ステップS15、ステップS16）。XML文書の場合は、XML文書を単純にツリー表示するスタイルシート、HTML文書の場合は、HTML文書をそのまま表示するスタイルシートがデフォルトスタイルである。これにより、一般的なXML文書やHTML文書についても、本システムでページ部品と

して取り扱うことができる。

【0135】

(ブロックWeb文書処理モジュール)

ブロックWeb文書処理モジュールM22は、ページ読込モジュールM21によって入力されたブロックWeb文書を管理し、ユーザに当該文書を表示し、かつサブコンテンツのための便利なGUI操作を提供する、システムの核となるモジュールである。なお、ホストモジュールM1の説明でも述べたが、1つのブロックWeb文書処理モジュールM22が管理するブロックWeb文書は1文書である。複数ページ(複数文書)を管理する場合には、ページ数だけブロックWeb文書処理モジュールM22を生成する必要がある。

【0136】

ブロックWeb文書処理モジュールM22は、複数のサブモジュールから構成されており、これらが互いに協調動作する。以下にそれぞれに焦点を当てて詳細説明する。

【0137】

(ページ部品管理モジュール)

ページ部品管理モジュールM221は、ページを含み、ページ内の全てのページ部品を管理し、構成変更のためのAPI(application programming interface)を提供する他のモジュール向けのベースサービスモジュールである。ページ部品のデータ構成は図4で説明したとおりである。

【0138】

図10に示すように、ページ部品管理モジュールM221は、ページ部品読込モジュールM221a、ページ部品データベース(DB)M221b、ページ部品挿入モジュールM221c、ページ部品削除モジュールM221d、ページ部品保存モジュールM221eという5つのサブモジュールから構成されている。

【0139】

ページ部品読込モジュールM221aは、入力されたブロックWeb文書を解析し、当該ブロックWeb文書中に含まれているサブコンテンツに対応する部分文書についてのページ部品を作成し、最終的に出来上がったページ部品群をペー

ジ部品DBM221bに登録する。

【0140】

ブロックWeb文書では、独立的に取り扱うサブコンテンツは、ブロックWeb文書そのものであったり、当該Web文書中の「component」属性の付与されているエレメント（要素）をトップノードとする部分文書に含まれるものであったり、あるいは、「import」エレメントで例えばURLで指定された外部リソースである。

【0141】

そこでページ部品読込モジュールM221aは、まず入力されたブロックWeb文書から、「component」属性の付与されている部分文書、「import」エレメントで構成される部分文書を抽出する。ただし、サブコンテンツ自体（サブコンテンツの実体）は、「component」属性が与えられている部分文書は、そのトップノードとしてのエレメントの子エレメントに含まれている。また、「import」エレメントの場合は、当該エレメントの「href」属性で指定されている外部リソースとしてのコンテンツを、公知技術を用いてダウンロードすることにより、サブコンテンツの実体（内容）を取得することができる。

【0142】

そして、ブロックWeb文書から抽出された各サブコンテンツについて、「component」属性が付与されているエレメントをトップノードとする部分文書、「import」エレメントから構成される部分文書のそれぞれを基に、図4に示したようなページ部品D1を生成する。なお、サブコンテンツとしてのブロックWeb文書自体についてのページ部品D1も生成されるが、その生成も上記同様である。

【0143】

図4のページ部品D1中のコンテンツD12には、例えば、サブコンテンツ（表示データ）が書き込まれる。

【0144】

各ページ部品を生成する際には、ブロックWeb文書で表されている部分文書

間の親子関係（XML文書中のエレメント間の親子関係と同様）から、各ページ部品間の親子関係が得られているので、親子関係の存在するページ部品については、それに従い、親のページ部品へ当該ページ部品を関連付けるためのポインタやリンクなどの情報（すなわち、ここでは、親ページ部品への参照D15）や、子供のページ部品を当該ページを関連付けるための情報（すなわち、ここでは子ページ部品管理情報D16）として、子ページ部品を当該ページ部品に関連付けるためのポインタやリンクなどの情報（子ページ部品への参照D162）や、当該ページ部品中の当該子ページ部品を挿入する場所（位置）D161を書き込む。

## 【0145】

各サブコンテンツについて、当該サブコンテンツに対応する部分文書が含まれているブロックWeb文書の文書構造上の位置を指定するポインタ情報であるXPointerつきURLは、リンクURLD17に書き込む。「import」エレメントによって外部から取得されるサブコンテンツの場合は、「import」エレメントのhref属性で指定されていたXPointerつきURLをリンクURLD17に書き込む。

## 【0146】

「component」属性を有する部分文書、「import」エレメントから構成される部分文書のそれぞれについて、その属性として、「type」属性が記述されていれば、その値を、タイプ名D11として書き込み、「style」属性が記述されていれば（その値がセミコロンで複数の文字列（各文字列は、例えばスタイルシートの識別子に対応する）に分割されていれば、その各文字列を）スタイルシートD13としてそれぞれ書き込む。

## 【0147】

「component」属性を有する部分文書、「import」エレメントを有する部分文書のそれぞれについて、例えば、当該部分文書中の子エレメントとして、「accept」エレメントが存在すれば、それは、当該部分文書に対応するサブコンテンツには、挿入可能なサブコンテンツがあることを表し、その挿入可能（受け入れ可能）なサブコンテンツのタイプ名が属性として示されてい

る。この場合、サブコンテンツを挿入するための受理情報D14を書き込む。まず、サブコンテンツの挿入場所（位置）D141として、当該「accept」エレメントの構造上の位置を書き込み、「accept」エレメントに「type」属性があれば（その値がセミコロンで複数の文字列（各文字列は、例えば受け入れ可能なサブコンテンツのタイプ（受理タイプ）の識別子に対応する）に分割されていれば、その各文字列を）受理タイプD142としてそれぞれ書き込む。

## 【0148】

「accept」エレメントは複数存在する可能性があり、その数だけ書き込み処理を行う。

## 【0149】

ページ部品DBM221bは、ブロックWeb文書に含まれるページ部品をすべて格納し管理している。各ページ部品は、自分の親となるページ部品および子供のページ部品群への参照情報を持っているので、この参照情報を基に順にたどっていくことで、任意のページ部品の情報を取り出すことができる。

## 【0150】

ページ部品挿入モジュールM221cは、任意のページ部品の子供として、新規にページ部品を追加・挿入するためのものである。

## 【0151】

図11は、ページ部品挿入モジュールM221cの処理動作を説明するためのフローチャートである。以下、図11を参照して説明する。例えば、ページ部品Aをページ部品Bに挿入する場合（ステップS20）、ページ部品DBからページ部品Bの受理情報D14に含まれる受理タイプD142を取出すとともに、挿入するページ部品Aに含まれているタイプ名D11を取出し、当該受理タイプD142に、タイプ名D11として書き込まれているページ部品Aのタイプ名が存在するか否かを判定する（ステップS20～23）。存在するときは、ページ部品Aをページ部品Bの子供のページ部品（子ページ部品）として関連付けるために、ページ部品Aについての子ページ部品管理情報D16をページ部品Bに登録する。また、ページ部品Bをページ部品Aの親のページ部品（親ページ部品）と

して関連付けるために、ページ部品Bについての親ページ部品への参照D15をページ部品Aに登録する（ステップS24）。

【0152】

ステップS23において、受理タイプD142に、タイプ名D11として書き込まれているページ部品Aのタイプ名が存在しないときは、当該ページ部品Aを適切なタイプのページ部品に変換をしてから挿入することを実現するため、ページ部品形式調整モジュールM225を呼び出す。その際、ページ部品形式調整モジュールM225にページ部品Bの受理タイプD142と、ページ部品Aを渡して、ページ部品Aのサブコンテンツのタイプをページ部品Bが受け入れ可能なタイプになるよう、変換を依頼する。

【0153】

ページ部品Aのサブコンテンツのタイプが、ページ部品Bが受け入れ可能なタイプに変換できたときには、ステップS24へ進む。

【0154】

なお、ステップS22で、ページ部品Aの受理情報D14に受理タイプが指定されていないとき、ステップS25で、ページ部品Aのサブコンテンツのタイプをページ部品Bが受け入れ可能なタイプに変換できないとき、ステップS26へ進み、ページ部品Aのページ部品Bへの挿入は拒否される。

【0155】

図10の説明に戻り、ページ部品削除モジュールM221dは、任意のページ部品を削除するためのものである。具体的には、削除されるページ部品の親ページ部品からは、当該削除されるページ部品を関連付けるための子ページ部品管理情報D16を削除し、当該削除されるページ部品自体をページ部品DBから削除する。このとき、削除対象のページ部品に対応する部分文書に、当該部分文書に包含される部分文書（すなわち、子供の部分文書）があるときは、当該包含される部分文書に対応するページ部品もページ部品DBから削除する。なお、ページ部品挿入モジュールの機能と、このページ部品削除モジュールの機能を組み合わせることで、ページ部品の移動も実現できる。

【0156】

ページ部品保存モジュールM221eは、任意のページ部品をファイル保存する機能である。指定されたページ部品を保存する機能であり、この機能を利用する場合には、保存したいページ部品とファイル名を指定する。ただし保存対象には、指定されたページ部品だけでなく、その子孫ページ部品も含まれる。

## 【0157】

保存の形式としては、ブロックWeb文書形式、PureXML形式、HTML形式の3種類ある。ブロックWeb文書形式とは、「component」属性や「accept」エレメントや「import」エレメントなど、ブロック化・リンクのための特殊な表記を含む形式であり、次回に、ページ部品記憶DBにロードしてもブロック構造情報やリンク関係を復元できる。

## 【0158】

PureXML形式は、これらブロック化のための特殊な表記を取り除いたものである。HTML形式は、現在表示されているHTMLの状態をそのまま保存するものである。

## 【0159】

ブロックWeb文書形式とPureXML形式の場合は、ページ部品D1のコンテンツD12に対して、当該ページ部品の子孫ページ部品のコンテンツD12を合成することで、保存内容を決定する。ページ部品D1には、子ページ部品管理情報D16として、各子ページ部品とその子ページ部品を挿入すべき挿入場所D161書かれている。コンテンツの合成の際には、この情報を用いる。

## 【0160】

一方、HTMLの場合は、後述するHTML合成モジュールM222a（図12参照）を実行することで、保存内容を決定する。

## 【0161】

（レンダリング処理モジュール）

レンダリング処理モジュールM222は、ページ部品を表示する機能を持つ。

## 【0162】

重要な用途の1つとして、ページ部品管理モジュールM221によって解析処理済のブロックWeb文書を図6に示すように、表示画面上のWeb表示部6-

2に表示する。

【0163】

ページ部品は、前述したように、他のページ部品に包含されることもあれば、他のページ部品を包含することもある（ページ部品は、必然的に他のページ部品と包含関係を有する）。各ページ部品に対応するサブコンテンツに対して、そのページ部品のスタイルシートD12を適用するだけでは不十分で、そのページ部品の各子ページ部品のスタイルシート適用結果を、はめ込む処理が必要となる。これに対して、本システムでは、スタイルシートに「import」エレメントを用意しており、「import」エレメントの「type」属性として挿入対象とする子ページ部品のタイプ名を書くとする（省略した場合はどんなタイプも受け入れる）。XSLT（スタイルシート）の適用処理をただけでは、「import」命令はそのままHTMLと混じって出力として出てくることになる。そこで、そのHTMLを解析しHTML合成を行う。

【0164】

なお、このスタイルシート中の「import」エレメント（「import」と呼ぶこともある）は、画面上に表示するためのHTML合成のためのものであり、ブロックWeb文書で記述されている「import」とは機能面で似てはいるが、別の命令である。

【0165】

具体例を挙げると、適用結果が以下のようなスタイルシートがあるとする。

【0166】

```
<html><body>
  <h1>Title</h1>
  <div><import type="type-a"></div>
  <div><import type="type-b"></div>
</body></html>
```

このとき、1番目の「div」エレメントには「type-a」タイプの子ページ部品のスタイル適用結果が入り、2番目の「div」エレメントには「ty



p e - b」タイプの子ページ部品のスタイル適用結果が入る。

【0167】

この合成処理をするのがHTML合成モジュールM222aである。HTML合成モジュールM222aは、指定されたページ部品に対してスタイルシート（複数のスタイルシートがある場合は、指定されたスタイルシートあるいは1番目）を適用してHTMLを得る。

【0168】

得られたHTMLの中から「i m p o r t」エレメントをすべて探し出し、それぞれに関して「s r c」属性で指定された子ページ部品に関して、スタイルシートを適用して得られたHTMLの結果を「i m p o r t」エレメントの代わりにはめ込む。ただし子ページ部品のスタイル適用結果にも「i m p o r t」命令が含まれていた場合には、はめ込む前に同様にその子ページ部品のスタイル適用結果をはめ込む。この処理は再帰的に行われる。

【0169】

この再起処理が完了すると、すべての子ページ部品のスタイルシート適用結果が合成された1つのHTMLを得られる。この結果をHTMLレンダラM222bに渡すことでHTMLが表示される。なお、この手順において使用されるHTMLレンダラM222bは入力されたHTMLを表示する機能を持つ既存のコンポーネントであり、一般のブラウザも内蔵しているものである。

【0170】

以上がブロックWeb文書を表示するまでの処理であるが、説明簡便化のため省略した事項がある。上記、各ページ部品あるいはその子ページ部品に関してスタイルシート適用をしてHTMLを得た直後に、得られたHTMLのルートエレメントおよびページ部品への参照を入力として、GUI処理モジュールM223のGUI処理初期化モジュールM223aを呼び出す。これは、合成されたHTMLにおいて、サブコンテンツのためのドラッグドロップやコンテキストメニューなどのGUI操作機能を埋め込むためである。詳しくは、後述する。

【0171】

（GUI処理モジュール）

GUI 処理モジュール M223 は、表示されたページ部品に対して、ドラッグドロップやコンテキストメニューなどのサブコンテンツ向け GUI 操作機能を提供するモジュールであり、図 13 のような構成になっている。

【0172】

GUI 処理初期化モジュール M223a は、後述するドラッグドロップ処理モジュール M223b やコンテキストメニュー処理モジュール M223c に対して、（マウス等で選択操作のなされた）処理対象の HTML のエレメントやページ部品を登録する処理を行う。

【0173】

GUI 処理初期化モジュール M223a は、ページ部品への参照情報、およびページ部品が表示において割り当てられている HTML 文書の要素（エレメント）であるルートエレメントへの参照情報の 2 つを入力として、呼び出されて、それらをドラッグ処理モジュール M223b、ドロップ処理モジュール M223c とコンテキストメニュー処理モジュール M223d にイベント登録することで、イベントと対応する処理を結びつける。

【0174】

これにより、登録された HTML エレメントでドラッグドロップ操作やコンテキストメニュー（右クリック）操作が行われたときに、これらの処理モジュールが呼び出されるようになり、サブコンテンツ向けのサービスが機能するようになる。

【0175】

また、これらの処理モジュールは、HTML エレメントとページ部品の対応表を持っているので、対象としているページ部品を特定できる。したがってページ部品の内容を基に、GUI サービス機能を提供することができる。

【0176】

以下はそれらイベントハンドラーの処理について述べる。

【0177】

ドラッグ処理モジュール M223b は、1 つのページ部品に対応する HTML 部分文書の表示領域がドラッグされたときに実行される処理モジュールで、一般

にドラッグイベントハンドラーとも呼ばれる。ユーザは、ドラッグドロップGUI操作によってページ部品を移動あるいはコピーさせることができる。ドラッグ処理モジュールはその前半部を担当する。

## 【 0 1 7 8 】

この処理手順は、図 1 4 に示すように、画面上で選択された表示データに対応する部分文書に対し、ドラッグ操作がなされると、操作対象の当該表示データに対応する部品データをメモリに一時記憶する（ステップ S 3 0 からステップ S 3 1）。このようにして、続いて発生するドロップイベントに備える処理を行う。補足すると、このドロップ先は、同じページ内とは限らなくて、別のプロセスのブロック Web 文書であっても、メーラー（電子メールソフト）やOAソフトなど別のアプリケーションであってもよいし、デスクトップであってもよい。そこで、

ステップ S 3 2 において、エクスポートモジュール M 2 2 4 を呼び出して、外部アプリケーション向けに当該ページ部品を転送するために、当該ページ部品をエクスポートモジュール M 2 2 4 へ渡す。

## 【 0 1 7 9 】

続いて、ドロップ処理モジュール M 2 2 3 c は、ページ部品上に何かがドロップされたときに実行される処理モジュールで、一般にドロップイベントハンドラーとも呼ばれる。ドラッグドロップGUI操作の後半部を担当する。この処理手順の詳細は図 1 5 に示す通りであって、サブコンテンツを操作対象とするドロップイベントが発生すると、当該操作対象のサブコンテンツに対応するページ部品をメモリから読み出して、ページ部品挿入モジュール M 2 2 1 c を用いて、ドロップ操作により指定された位置にサブコンテンツを挿入する（ステップ S 4 0 ～ 4 2）。挿入されると、当該ドロップされたサブコンテンツに対応するページ部品を新規に作成する。先に読み出された、当該操作対象のサブコンテンツに対応するページ部品中の親ページ部品への参照情報 D 1 5 を参照して、当該操作対象のページ部品は、同じページからドラッグされたものであるか否かを判定する。同じページからドラッグされたものであるならば、当該操作対象のページ部品をページ部品削除モジュール M 2 2 1 d を用いてページ部品 D B M 2 2 1 b から削

除する（ステップS43～ステップS45）。そして、レンダリング処理モジュールM222を用いて、再描画を試みる（ステップS46）。一方、ステップS44で、操作対象のページ部品が同じページからドラッグされたものでないならば、そのままステップS46へ進む。

## 【0180】

補足として、マウス右ボタンを押しながらのドロップであった場合は、ユーザはリンクを意図していると解釈し、リンクであることを示すフラグ（図4のリンクフラグD18）を立てて（「1」にセットする）。この場合、図15のステップS42とステップS43の間で、右ドロップかどうかを判定するステップを追加し、右ドロップであればリンクフラグD18を「1」にセットし、そうでなければリンクフラグD18を「0」とする。また、この場合、リンクURLD17に、当該リンク情報を格納してもよい

なお、リンクとしておく場合、リンク先のページ部品のサブコンテンツが変化した場合でも、ブロックWeb文書を表示する際に、リンク先へ当該ページ部品を取りに行くため、その最新版のサブコンテンツが表示されるというメリットがある。たとえば、ユーザは、後述するスクラップブックに、所望するページ部品へのリンクを登録しておくことで、例えば、あるブロックWeb文書中の「CD売り上げベスト10」というサブコンテンツと、他のブロックWeb文書中の「図書売り上げベスト10」というサブコンテンツの両方が掲載された、毎日変化する自分だけのマイオリジナルポータルページを作成することができる。

## 【0181】

コンテキストメニュー処理モジュールM223dは、ページ部品上で右クリックされたときに、実行可能なコマンドのメニューをポップアップする機能である。本システムでは、コンテキストメニューとして削除コマンド・編集コマンド・新規作成コマンド・HTMLカッターコマンド・スタイル切り替えコマンドを用意している。削除コマンドが選択されると、ページ部品削除モジュールM211dを実行した上で、レンダリング処理モジュールM222が実行されて、ページ部品の削除が行われる。

## 【0182】

また編集コマンドと新規作成コマンドは、既存あるいは新規のページ部品を編集するコマンドであり、これらの処理は編集モジュールM223eが担当する。

## 【0183】

具体的な処理の流れを、図20に示す。図20において、たとえば、図6に示したWeb表示部6-2上の所望のサブコンテンツを選択した後、新規作成コマンドが選択されると、空のページ部品を作成して、この空のページ部品を、ページ部品挿入モジュールを用いて、当該選択されたサブコンテンツに対応するページ部品に挿入する（ステップS60～ステップS62）。その後、XML文書やXSLT文書を編集するためのXML/XSLEディタを起動して、各種編集コマンドを用いることで、新規作成されたページ部品中のコンテンツD12やスタイルシートD13の内容（実体）を作成する（ステップS63）。編集作業が終了し、XML/XSLEディタが閉じられたら、編集結果を先に新規作成されたページ部品中のコンテンツD12やスタイルシートD13に書き込む（ステップS64）。

## 【0184】

ただし、ステップS63で起動されるXML/XSLEディタは既存のものでよい。

## 【0185】

HTMLカッターコマンドは、入力ページがもともと通常のHTMLである場合に（この場合、当該HTML文書は、ページ読込モジュールM21において、それ全体が1つの部品データとなるようなブロックWeb文書に変換されている）、図6のWeb表示部6-2上でユーザが範囲指定をし、その部分をページ部品として分割する機能であり、HTMLカッターモジュールM223dが担当する。ページ部品として分割すると、その部分はその後、独立したサブコンテンツとして取り扱うことができる。HTMLカッターの具体的な処理の流れを図21に示す。

## 【0186】

図21において、たとえば、文書処理モジュールM2に入力した時点では、通常のHTML文書であったものが、ページ読込モジュールM21を経て、全体が

1つの部品データとなるようなブロックWeb文書に変換され、さらに、ページ部品管理モジュールM221、レンダリング処理モジュールM222などを経由して、図6に示したWeb表示部6-2上に表示されたとする。このWeb表示部6-2上で、ユーザにより所望の領域が選択された後、HTMLカッターコマンドが選択されると、まず、現在の選択範囲内におけるトップエレメントを特定する（ステップS70～ステップS72）。

## 【0187】

特定されたエレメントに、当該エレメントは1つのサブコンテンツであることを示すための「component」属性と、スタイル属性を付加する。すなわち、当該エレメントに「component = "yes"」と、スタイル属性として、HTML用のデフォルトスタイルが記述されているスタイルシートの識別情報を書き込む（ステップS73）。選択範囲内のコンテンツ、すなわち、上記トップエレメント以下の選択範囲内のエレメント（要素）から構成される部分文書に対するページ部品を新規作成し、ページ部品DBへ登録する（ステップS74）。

## 【0188】

このサブコンテンツとして指定されたコンテンツを含む、元のコンテンツ（1つのブロックWeb文書としての1つのページ）に対応するページ部品と、新規作成されたページ部品との間で親子関係があるので、この親子関係に基づき、前述したように、それぞれのページ部品中の親ページ部品への参照D15と、子ページ部品管理情報D16を書き込む（ステップS75）。

## 【0189】

補足であるが、ステップS71でのHTMLベースのページかどうかの判定は、スタイルシートがHTML用デフォルトスタイルであるかどうかで判定すればよい。もちろん、ブロックWeb文書や、それ以外の各種構造化文書である場合にも上記HTMLカッターコマンドと同等のコマンドを用意して、それら構造化文書（すなわち、部分文書）から所望の部分文書をサブコンテンツとして区分するようにいしてもよい。

## 【0190】

スタイル切り替えコマンドは、ページ部品に登録されている複数のスタイルシートD 1 3の中から、ユーザに所望のものを選択させ、その選択されたスタイルシートでページ部品D 1を表示しなおす機能であり、スタイル切り替えモジュールM 2 2 3 fが担当する。スタイル切り替えの具体的な処理の流れを図2 2に示す。

#### 【0 1 9 1】

図2 2において、たとえば、図6に示したWeb表示部6-2上の所望のサブコンテンツを選択した後、スタイル選択コマンドが選択される（ステップS 8 0）。ステップS 8 1において、当該選択されたサブコンテンツに対応するページ部品D 1に複数のスタイルシートD 1 3が登録されているときには、ステップS 8 2へ進み、1つのみスタイルシートD 1が登録されているときには、そのまま処理を中止する。

#### 【0 1 9 2】

ステップS 8 2において、当該選択されたページ部品に登録されている複数のスタイルシートのうちのいずれか1つを選択させるためのメニュー画面を表示し、この中から、ユーザにより選択されたスタイルシートを、当該選択されたサブコンテンツ（ページ部品D 1中のコンテンツD 1 2）に適用して、Web表示部6-2に表示する（ステップS 8 3）。

#### 【0 1 9 3】

その他、説明は省略するが、切り取り、コピー、貼り付けなどのコマンドがあってもよい。

#### 【0 1 9 4】

（エクスポートモジュール）

図2のエクスポートモジュールM 2 2 4は、前述のように、文書処理モジュールM 2と外部アプリケーションとの間でサブコンテンツをドラッグドロップ操作により複写可能にするためのモジュールである。また、ここでは、説明は省略するが、文書処理モジュールM 2からドラッグドロップされたサブコンテンツを外部アプリケーションの形式でファイル保存する目的にも利用可能である。

#### 【0 1 9 5】

ここでエクスポートとは、文書処理モジュールM2から外部アプリケーションへのサブコンテンツの複製を転送することを意味する。

【0196】

また、外部アプリケーションとは、本実施形態に係る文書処理システム以外のアプリケーションを云うものとする。

【0197】

エクスポートモジュールM224は、図16のように複数のサブモジュールから構成されている。外部アプリケーションからは、ページ部品を入力とするエクスポート要求を受けるが、その要求は共有メモリ登録モジュールM224aに送られる。共有メモリ登録モジュールM224aは、他の出力モジュールM224bからM224dを用いて、処理対象のページ部品を転送するための当該ページ部品を含むエクスポートデータを作成させ、それらを全ての外部アプリケーションからアクセスすることのできる共有メモリに登録する。これにより、外部アプリケーションは、ページ部品がドロップされたときに、その外部アプリケーションにとって最も都合の良いデータ形式で内容を読み込むことができる。

【0198】

アプリ別形式出力モジュールM224bは、いくつかの著名な外部アプリケーションに対象を絞り、処理対象のページ部品（少なくとも当該ページ部品で表示する表示データ）を、それら外部アプリケーションに適用可能なデータ形式に変換することにより、上記エクスポートデータを生成し、当該外部アプリケーションへ出力する。

【0199】

標準形式出力モジュールM224cは、処理対象のページ部品（少なくとも当該ページ部品で表示する表示データ）を、テキスト形式やHTML形式など広く使われているデータ形式に変換することにより、上記エクスポートデータを生成し、外部アプリケーションへ出力する。

【0200】

オブジェクト埋込形式出力モジュールM224dは、処理対象のページ部品（少なくとも当該ページ部品で表示する表示データ）を、本システムのデータ形式



(すなわち、例えば、ページ部品D1のデータ形式)のまま、当該ページ部品に対応する文書処理モジュールM2を当該ページ部品に対応付けるための参照情報とともに、上記エクスポートデータとして外部アプリケーションへ出力する。この方法では、ドロップ先の外部アプリケーションのウィンドウ領域の一部に本システムの文書処理モジュールM2が埋め込まれる。

#### 【0201】

(ページ部品形式調整モジュール)

図2のページ部品形式調整モジュールM225は、前述のように、ページ部品挿入処理モジュールM221cで用いられるモジュールである。具体的には、ページ部品挿入処理モジュールM221cから、変換対象のページ部品と、他のページ部品中の受理タイプD142を含む変換要求が入力されると、当該変換対象のページ部品のタイプ名D11を調べ、少なくとも変換対象のページ部品に対応する部分文書のタイプ(例えば、データ形式)を他のページ部品中の受理タイプD142に含まれるいずれかのタイプに変換する処理である。

#### 【0202】

この変換要求は、ページ部品挿入処理モジュールM221cから変換ルール検索モジュールM225aに送られる。

#### 【0203】

変換ルールデータベース(DB)M225bには、変換元と変換先のタイプ別に、複数の変換方法(変換規則あるは変換ルール)が予め記憶されている。

#### 【0204】

変換ルール検索モジュールM225aは、変換ルールDBM225bから、変換元と変換先のタイプに対応する変換方法を見つけ出し、変換結果のページ部品を返すか、あるいは見つからなかった場合には、変換に失敗した旨を要求元のページ部品挿入処理モジュールM221cへ返す。

#### 【0205】

変換ルールDBM225bには、変換元のタイプと変換先のタイプと、変換ルール(例えばXSLT文書などの変換規則を記述した文書)の3つを1組とした複数のデータを管理している。変換ルール検索モジュールM225aは、入力さ

れた、処理対象のページ部品のタイプ名D11から、最終的に求めているタイプのいずれかに変換するための、複数の変換ルールからなるシーケンスパス（順序つき組み合わせ）を検索し、最終的に複数のシーケンスが見つかった場合には、最短パスとなる組み合わせを採用する。

## 【0206】

たとえば、処理対象のページ部品のタイプが「type-x」タイプであり、最終目的のタイプが「type-y」タイプか「type-z」タイプであり、変換ルールDBM225bには、以下のデータが記憶されていたとする。

## 【0207】

(type-x, type-a, xa.xml)

(type-a, type-b, ab.xml)

(type-b, type-y, by.xml)

(type-x, type-c, xc.xml)

(type-c, type-z, cz.xml)

この場合、最終目的のタイプに変換するシーケンスとして、

xa.xml → ab.xml → by.xml、と、

xc.xml → cz.xml、の2種類があるが、変換ルール検索モジュールM225aは後者を選択し、順にXSLT文書を適用した結果得られるページ部品（「type-z」タイプ）を返す。

## 【0208】

（デスクトップ貼り付けモジュール）

デスクトップ貼り付けモジュールM3は、独立したプロセスで動作するアプリケーションである。このアプリケーションソフトウェアにより、一般的なブラウザで構成されている第1のホストモジュールM1や第2のホストモジュールM1を組み込んだアプリケーションソフトウェアからデスクトップへページ部品（サブコンテンツ）をドラッグドロップすると、当該ページ部品を付箋紙のようにデスクトップに貼り付けることが可能となる。

## 【0209】

また、逆に付箋紙からホストモジュールM1へドラッグドロップして戻すこと

も可能となっている。

【0210】

デスクトップ貼り付けモジュールM3は図18に示すように、デスクトップ監視モジュールM3aと付箋紙表示モジュールM3bから構成されている。付箋紙表示モジュールM3bは付箋紙の数だけ動的に生成されるモジュールである。また各付箋紙表示モジュールM3bは、内部に文書処理モジュールM22を生成維持する。

【0211】

また、デスクトップ貼り付けモジュールM3において、ページ部品のデスクトップへのドロップイベントに関する処理の流れを図19に示す。

【0212】

図19において、たとえば、図6に示したWeb表示部6-2上の所望のサブコンテンツを選択した後、ドラッグ操作と、それに引き続き、デスクトップ上にドロップイベントが発生したことをデスクトップ監視モジュールM3aが検知すると（ステップS50）、付箋紙表示モジュールM3bが起動する（ステップS51）。このとき、選択されたものがサブコンテンツ（ページ部品）でないときは、処理を中止する（ステップS56）。

【0213】

起動された付箋紙表示モジュールM3bは、画面上にウィンドウ領域を確保するとともに、文書処理モジュールM2を起動する（ステップS52からステップS54）。

【0214】

付箋紙表示モジュールM3bは、起動した文書処理モジュールM2にドロップされたサブコンテンツ（ページ部品）を読み込ませ、当該ウィンドウに表示させる（ステップS55）。

【0215】

これにより、ページ部品を付箋紙に次々貼り付けることができる。なお公知の技術なのでここでは方法について言及していないが、各付箋紙表示モジュールが作成したウィンドウは、後から画面上のユーザの好きなところに移動させることが

できるようにする。

【0216】

逆に付箋紙として表示されたページ部品は、逆にブラウザヘドラッグドロップすることで、ブラウザが表示しているページへ追加したり、あるいは削除することができる。

【0217】

これらの機能はデスクトップ貼り付けモジュールM3が特別に用意した機能ではなく、以前の詳細説明した各付箋紙表示モジュールM3b内の文書処理モジュールM2.2が行っていることである。

【0218】

(文書編集システムのまとめ)

以上説明したように、本実施形態に係る文書編集システムは、複数の要素から構成された文書構造を有する構造化文書を表示手段(図24の出力装置24に対応する)に表示するとともに、この構造化文書に一致または含まれる部分文書であって、予め操作単位として定められた表示データを表示手段に表示するための少なくとも1つの要素からなる部分文書を基に構造化文書を編集する文書編集システムであって、部分文書には、少なくとも1つのサブコンテンツに対応する部分文書であることを表す情報(例えば、「component」属性)が含まれており、この情報を基に、少なくとも当該部分文書の位置情報(部分文書間の親子関係(包含関係)を表す情報を含む)を含む部分文書毎の部品データ(すなわち、ページ部品)を作成して、構造化文書を部品データの集合として記憶手段(すなわち、ページ部品DB)に記憶し、表示手段に表示された表示データのうち、操作対象として選択された表示データに対する操作内容に応じて、当該操作対象の表示データに対応する部品データを更新することにより構造化文書を編集する。

【0219】

各部分文書には、当該部分文書のタイプと、当該部分文書のデータ形式を当該部分文書に適用する表示形式に対応するデータ形式に変換するための少なくとも1つの変換規則と、当該部分文書の表示データとして表示手段に表示される他の

構造化文書中の部分文書へのリンク情報と、当該部分文書に挿入可能な他の部分文書のタイプと挿入位置のうちの少なくとも1つが、当該部分文書の付加情報（例えば、属性情報など）として含まれており、この付加情報を各部分文書から抽出して、当該部分文書の部品データを作成する。従って、部品データには、当該部品データに対応する部分文書に含まれている付加情報が含まれている。

## 【0220】

なお、例えば、部分文書の位置は、当該部分文書を包含している、例えばWebページなどの他の部分文書の文書構造上の配置位置などで表すことができる。

## 【0221】

また、1つのWebページ全体に対応する部分文書と、当該Webページに含まれる1つのサブコンテンツに対応する部分文書との間の関係は、親子関係（包含関係）にあると言える。

## 【0222】

また、各部分文書で表示手段に表示されるデータ（表示データ）は、当該部分文書に包含されている場合と、当該表示データへのリンク情報が当該部分文書に含まれている場合とがある。

## 【0223】

このように、上記実施形態によれば、構造化文書に含まれている、操作単位の部分文書を表す情報を基に識別することができる各部分文書のそれぞれについて、その付加情報を基に、上記部品データ（ページ部品）を作成することで、この部品データを基に、画面上に表示された内容をサブコンテンツ単位（部分文書単位）で容易に編集操作を行うことができる。

## 【0224】

（サービスサーバ）

以上説明したように、ブロックWeb文書を用いたサブコンテンツ単位の操作・編集を行う仕組みが確立された場合、Webページの提供者（Webサイト）の中には、この仕組みを活用してページの利便性を向上させることで、当該Webページの閲覧者を増やしたいと考える者もいるだろう。

## 【0225】

Web ページを提供する側として、ポータルサイトで各ポートレットをページ部品とするなど静的なWeb ページへの適用例と、ショッピングサイトなどで検索結果の商品カタログをページ部品とするなど動的なWeb ページへの適用例が考えられる。すなわち、前者の場合、サーバで保持するWeb ページは全て、ブロック構造情報などを有するブロックWeb 文書とし、後者の場合、例えば、検索される各レコードを1つのサブコンテンツ（ページ部品）とし、検索結果として得られた複数のレコードを所定の形式でまとめ上げて、1つのブロックWeb 文書を作成し、クライアントへ配信するようにしてもよい。

## 【 0 2 2 6 】

Web サイトから提供されるWeb ページが前述のブロックWeb 文書であると、Web サイトから提供されるコンテンツをユーザがブラウザで閲覧すると、当該Web ページに含まれるページ部品に関する上述の操作機能を活用することができるという効果が得られる。

## 【 0 2 2 7 】

図 2 3 は、サーバ装置の構成例を示したもので、サーバ装置 1 0 は、例えば、インターネットなどの所定のネットワーク 1 2 を介して複数のクライアント端末（ここでは、2 つ）1 1 a、1 1 b（以下、これらを区別する必要がない場合は、それぞれをクライアント端末 1 1 と呼ぶ）からの要求を受け付けて、ユーザに所望のWeb 文書（Web ページ）などを配信するためのものである。

## 【 0 2 2 8 】

図 2 3 に示すサーバ装置 1 0 は、クライアント端末 1 1 からの配信要求を受け付ける配信要求受付部 1 0 a と、複数のWeb ページ（全てブロックWeb 文書であってもよいし、HTML 文書などのブロックWeb 文書以外の文書であってもよい）を記憶するWeb ページ記憶部 1 0 b と、ブロックWeb 文書以外のWeb 文書をブロックWeb 文書に変換するための変換部 1 0 c と、配信要求されたWeb ページを要求元のクライアント端末 1 1 へ配信するための配信部 1 0 d と、上記各部を制御するための制御部 1 0 e とから構成されている。

## 【 0 2 2 9 】

Web ページ記憶部 1 0 b には、前述したように、ブロックWeb 文書のみを

記憶するようにしてもよい。また、ブロックWeb文書以外のWeb文書が記憶されていてもよい。Webページ記憶部10aに記憶されているWebページのうち、ブロックWeb文書以外のものは、変換部10cにて、ブロックWeb文書に変換する。

#### 【0230】

変換部10cは、前述のページ読込モジュールM21で構成可能である。すなわち、Webページ記憶部10bに格納されているWeb文書に対し適用可能な変換ルールが存在すれば、当該変換ルールを適用して、ブロックWeb文書に変換し、適用可能な変換ルールが存在しない場合には、当該Web文書を1つのページ部品と解釈して、当該Web文書のルートノードに、「component」属性を付与するとともに、当該Web文書にデフォルトのスタイルを与える。

#### 【0231】

このようにして、配信要求受付部10aで受け付けられた、クライアント端末11からの配信要求を基に、クライアント端末11から要求されたブロックWeb文書が配信部10dを介して、要求元へ配信するようになっている。

#### 【0232】

ところで、本システムの別の特徴として、ユーザはWebサイトやWebページを越えて、ページ部品を受け渡すことができる。したがって、ユーザがコンテンツを提供するWebサイトからページ部品だけを抽出することができ、ユーザは好みのWebサイトをベースに情報閲覧作業を続けることができる。

#### 【0233】

そこで各Webサイトは、Webテンプレート領域、すなわちページ部品を収納している余白部分で差別化を図ることができる。差別化として、デザインや機能で使い勝手を向上させて顧客をインセンティブを与えると同時に、バナー広告を行ったり、検索結果として自社製品を効果的に出力するなど自社サービスへの誘導を行うことで、メリットを得ることができる。

#### 【0234】

システム構成としては、Webテンプレート領域だけを含むWeb文書・ページ部品を検索出力するCGI・検索CGIの出力結果とWebテンプレート領域

を合成出力するCGIをWebサーバ上に配置するというシステム構成となる。

【0235】

またサーバが提供する別のサービスとして、顧客（ユーザ）が、図2や図24に示した文書編集システムを用いて作成したスクラップブックをサーバが代行管理するというサービスも考えられる。スクラップブックは、ユーザの操作するコンピュータ上に格納されるため、外出先などそのマシンにアクセスできない場所では、自分がスクラップブックに蓄積したサブコンテンツをアクセスすることができないという不便さがある。この問題を解決するサービスとして、サーバがスクラップブックを管理するというソリューションが考えられる。この場合、サーバはユーザを認証し、そのユーザでしかログインできないURLに、そのユーザのスクラップブック群を割り当てて閲覧できるようにするとともに、スクラップブックを更新するためのフォームを用意する。

【0236】

またサーバは、ユーザに表示するスクラップブックを、自社にとって都合の良い内容にすることができるので、たとえば自社製品情報を各スクラップブックに付記するなどして宣伝効果を得ることができる。

【0237】

あるいは、スクラップブックに記載されたサブコンテンツの内容から、データマイニングなど既存技術などを用いて、各ユーザの趣向などを分析し、当該ユーザにとって歓迎されることが期待される情報を各スクラップブックに付記するという、付加価値サービスで差別化することも可能となる。

【0238】

本発明の実施の形態に記載した本発明の手法は、コンピュータに実行させることのできるプログラムとして、磁気ディスク（フロッピーディスク、ハードディスクなど）、光ディスク（CD-ROM、DVDなど）、半導体メモリなどの記録媒体に格納して、あるいはインターネットなどのネットワークを介して頒布することもできる。

【0239】

なお、本発明は、上記実施形態に限定されるものではなく、実施段階ではその



要旨を逸脱しない範囲で種々に変形することが可能である。さらに、上記実施形態には種々の段階の発明は含まれており、開示される複数の構成要件における適宜な組み合わせにより、種々の発明が抽出され得る。例えば、実施形態に示される全構成要件から幾つかの構成要件が削除されても、発明が解決しようとする課題の欄で述べた課題（の少なくとも1つ）が解決でき、発明の効果の欄で述べられている効果（の少なくとも1つ）が得られる場合には、この構成要件が削除された構成が発明として抽出され得る。

【 0 2 4 0 】

【発明の効果】

以上説明したように、本発明によれば、構造化文書の中に含まれるコンテンツを、それよりさらに細かい操作単位としてのサブコンテンツ単位で容易に編集することができる。

【図面の簡単な説明】

【図 1】

本発明の実施形態に係る文書編集システムの機能を説明するための図。

【図 2】

文書編集システムの機能的な構成例を示した図。

【図 3】

1つのWebページ中のページ部品（サブコンテンツあるいは部分文書）について説明するための図。

【図 4】

ページ部品のデータ構成を説明するための図。

【図 5】

ホストモジュールの機能的な構成例を示した図。

【図 6】

画面表示例を示した図。

【図 7】

ページ読込モジュールの機能的な構成例を示した図。

【図 8】

ページ読込モジュールの動作を説明するためのフローチャート。

【図 9】

入力文書のブロック Web 文書への変換手順を説明するためのフローチャート

【図 10】

ページ部品管理モジュールの機能的な構成例を示した図。

【図 11】

ページ部品挿入モジュールの動作を説明するためのフローチャート。

【図 12】

レンダリング処理モジュールの機能的な構成例を示した図。

【図 13】

GUI 処理モジュールの機能的な構成例を示した図。

【図 14】

ドラッグ処理モジュールの動作を説明するためのフローチャート。

【図 15】

ドロップ処理モジュールの動作を説明するためのフローチャート。

【図 16】

エクスポートモジュールの機能的な構成例を示した図。

【図 17】

ページ部品形式調整モジュールの機能的な構成例を示した図。

【図 18】

デスクトップ貼り付けモジュールの機能的な構成例を示した図。

【図 19】

デスクトップ貼り付けモジュールの動作を説明するためのフローチャート。

【図 20】

ページ部品の編集動作を説明するためのフローチャート。

【図 21】

HTML カッター処理動作を説明するためのフローチャート。

【図 22】

スタイル切り替え処理動作を説明するためのフローチャート。

【図 23】

サーバ装置の構成例を示した図。

【図 24】

図 2 に示した文書編集システムを例えばコンピュータ（計算機）上で実現する場合の構成例を示した図。

【符号の説明】

M1 … ホストモジュール

M2 … 文書処理モジュール

M21 … ページ読込モジュール

M22 … ブロック Web 文書処理モジュール

M221 … ページ部品管理モジュール

M222 … レンダリング処理モジュール

M223 … GUI 処理モジュール

M224 … エクスポートモジュール

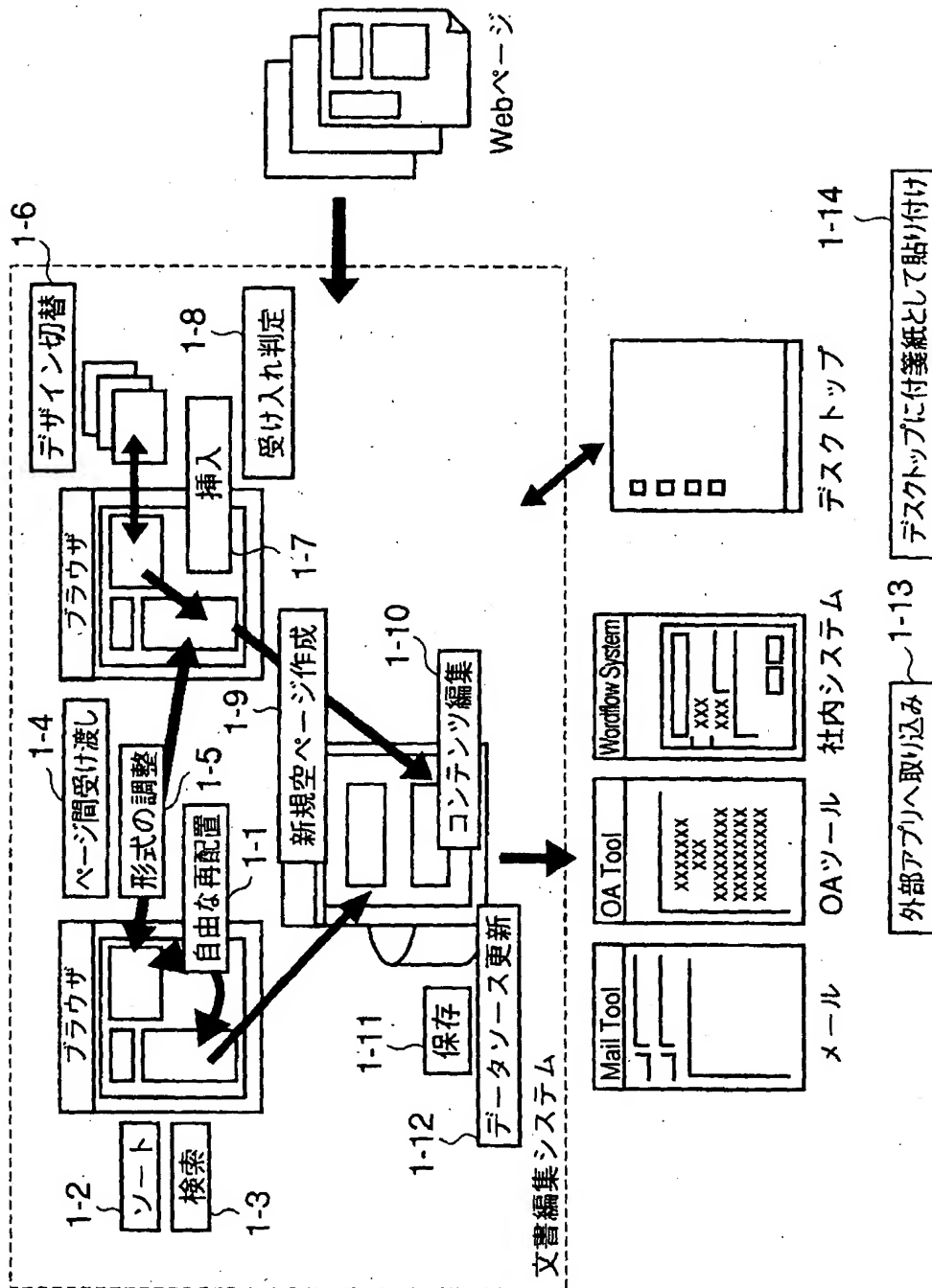
M225 … ページ部品形式調整モジュール

M3 … デスクトップ貼り付けモジュール

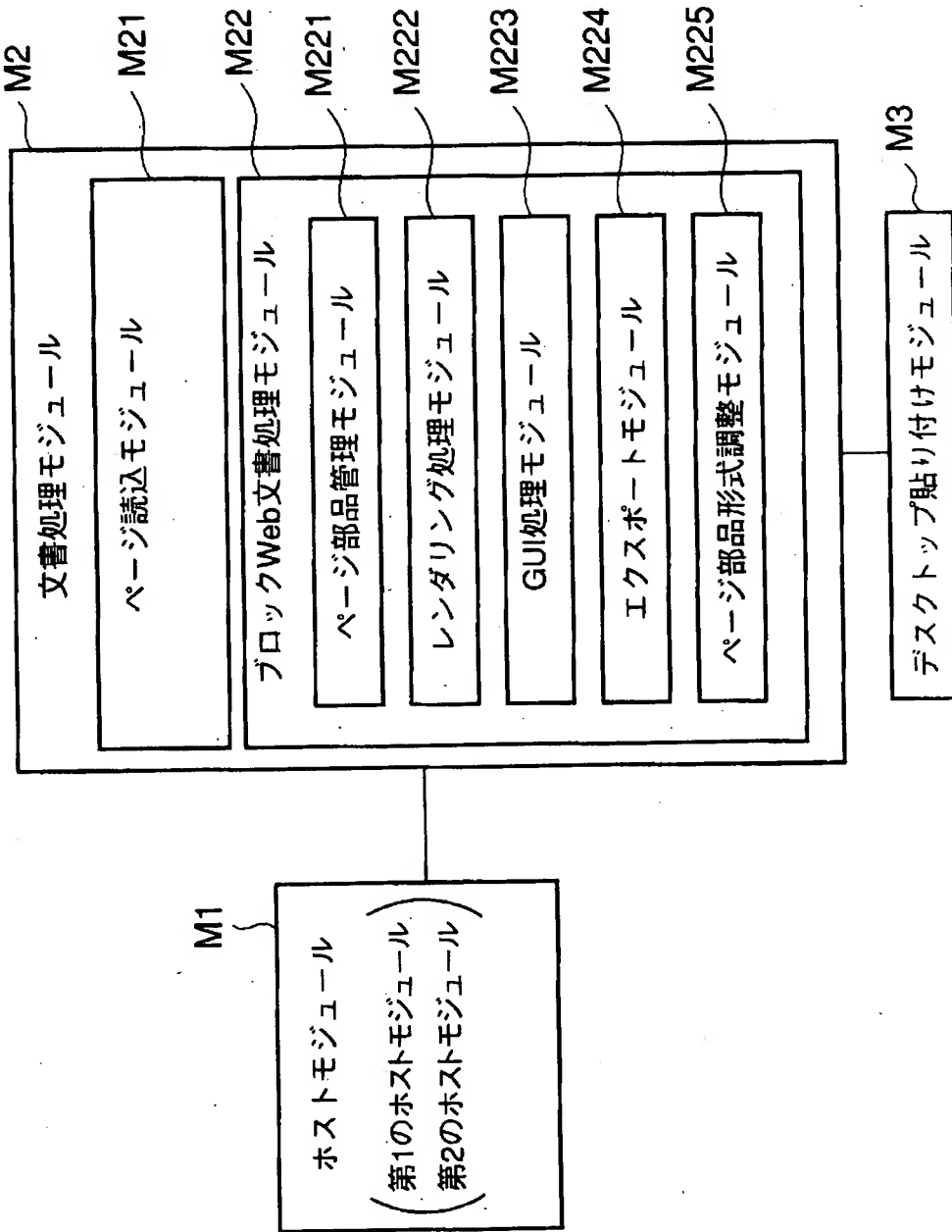
【書類名】

図面

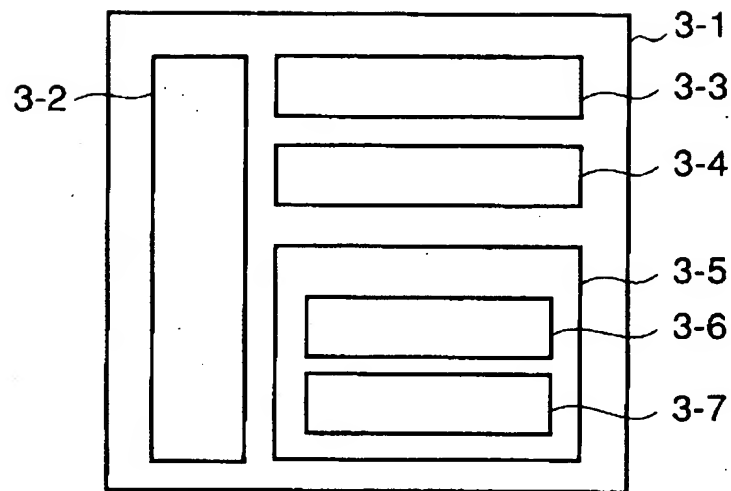
【図1】



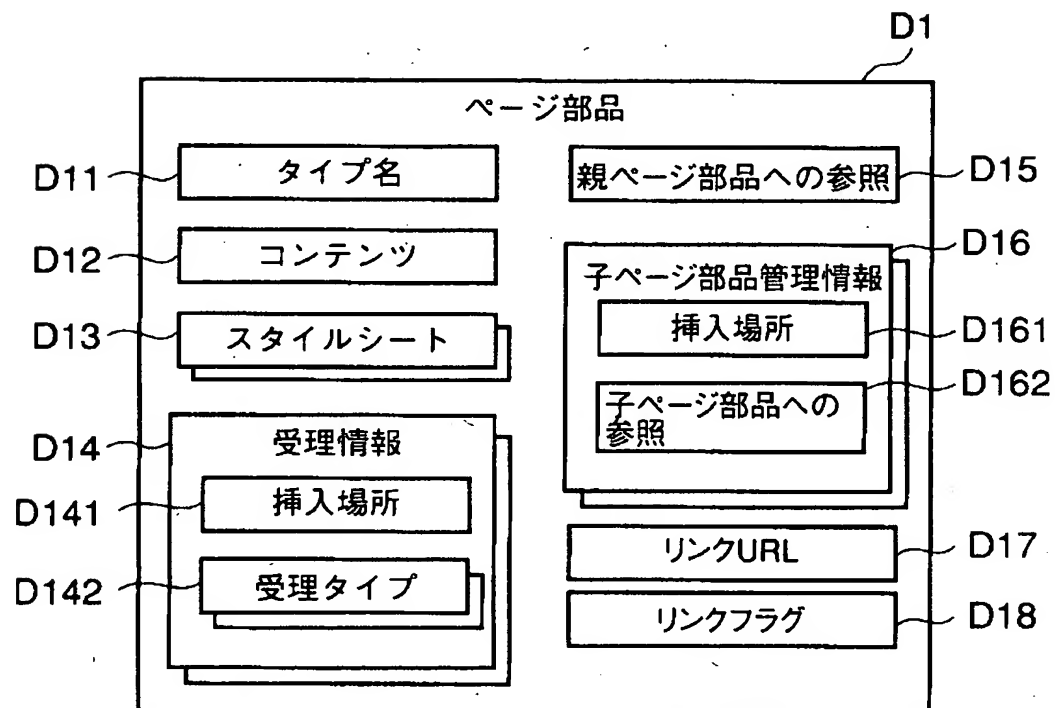
【図2】



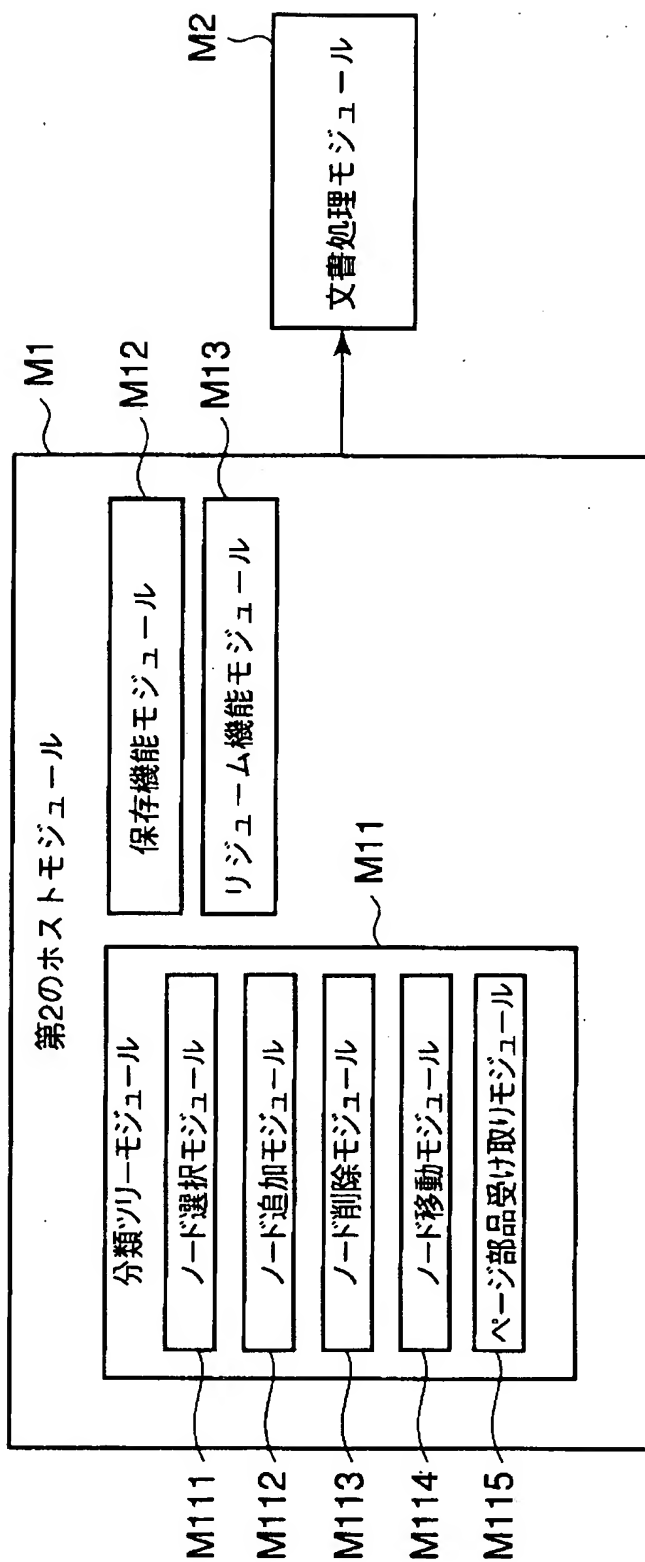
【図3】



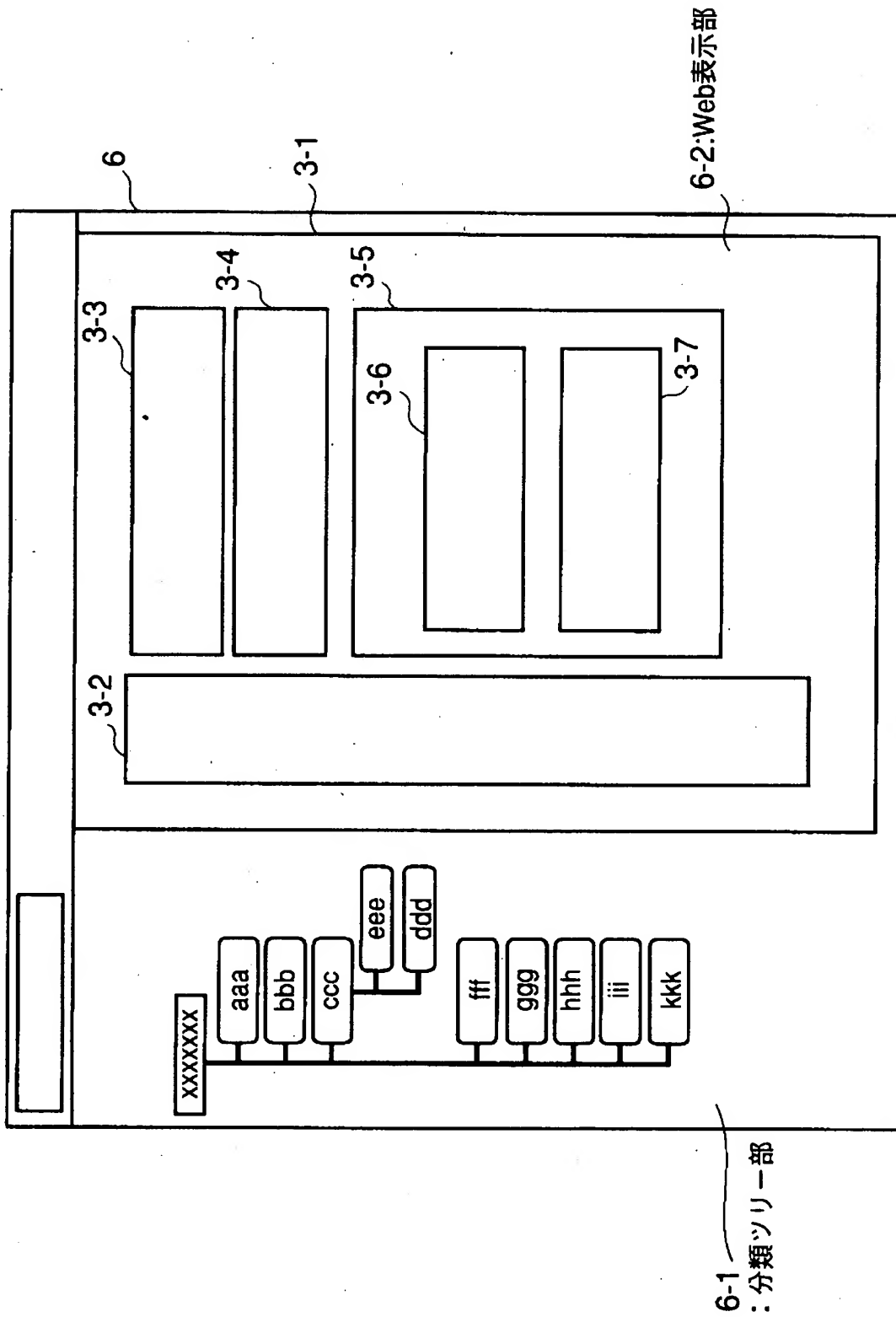
【図4】



【図 5】

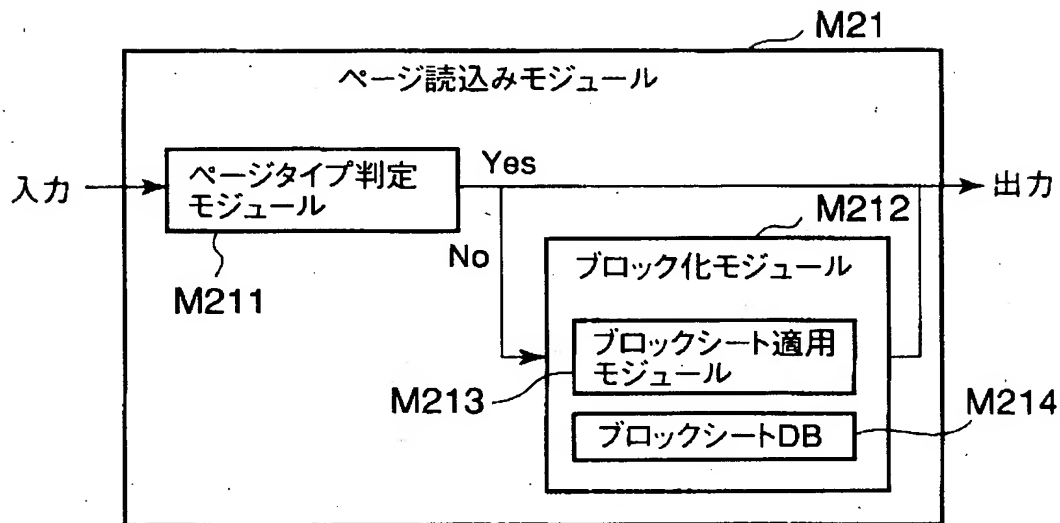


【図 6】

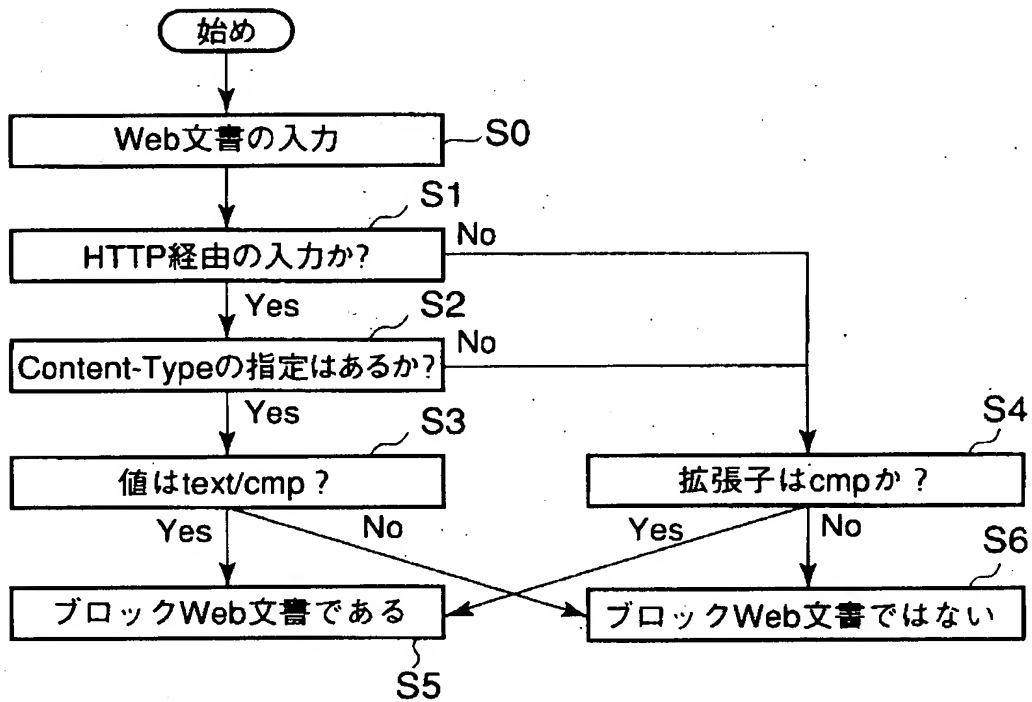




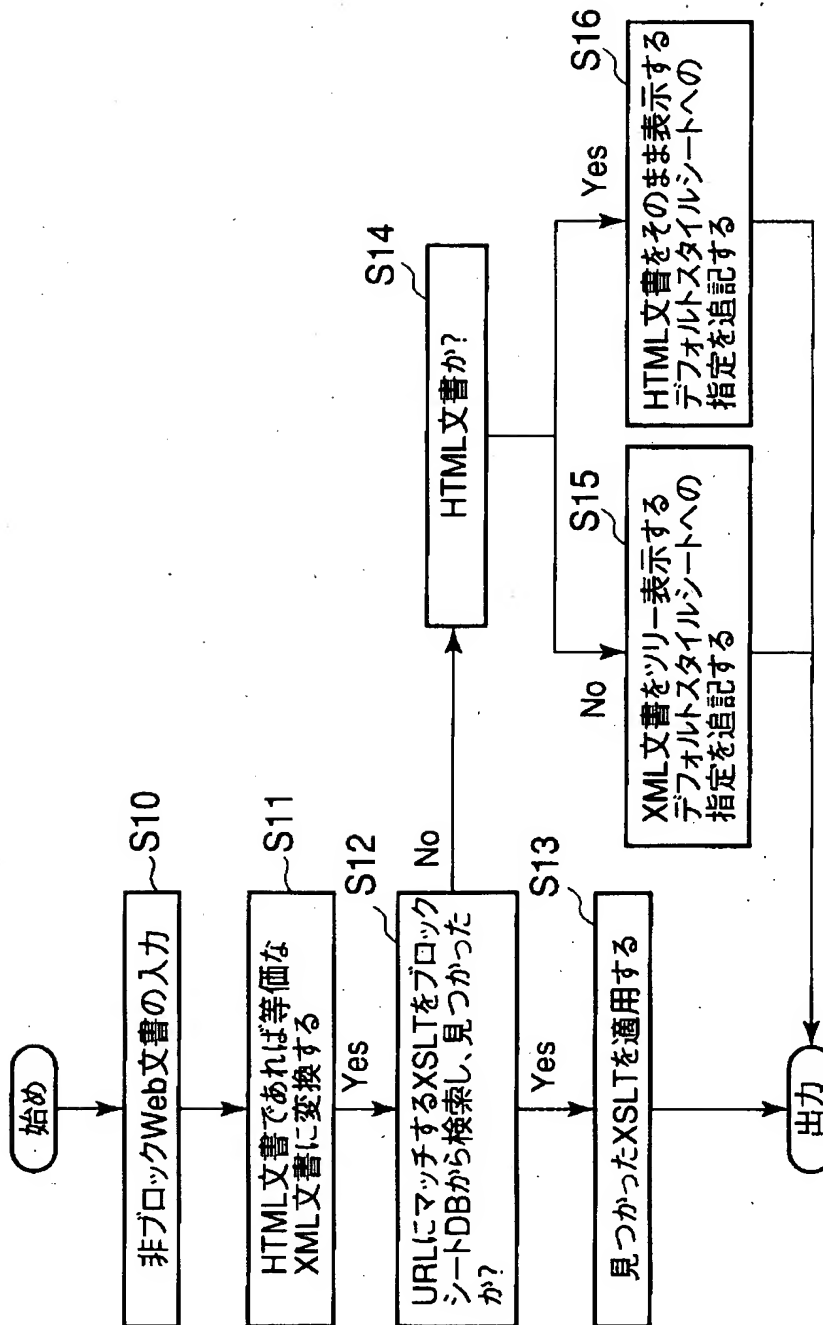
【図 7】



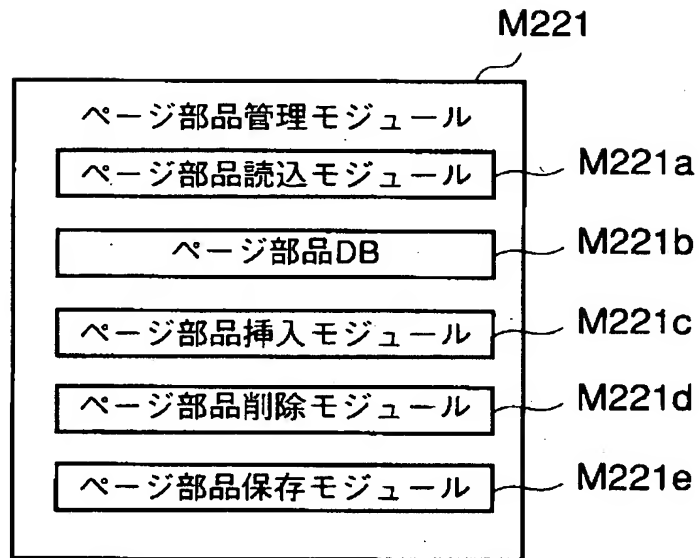
【図 8】



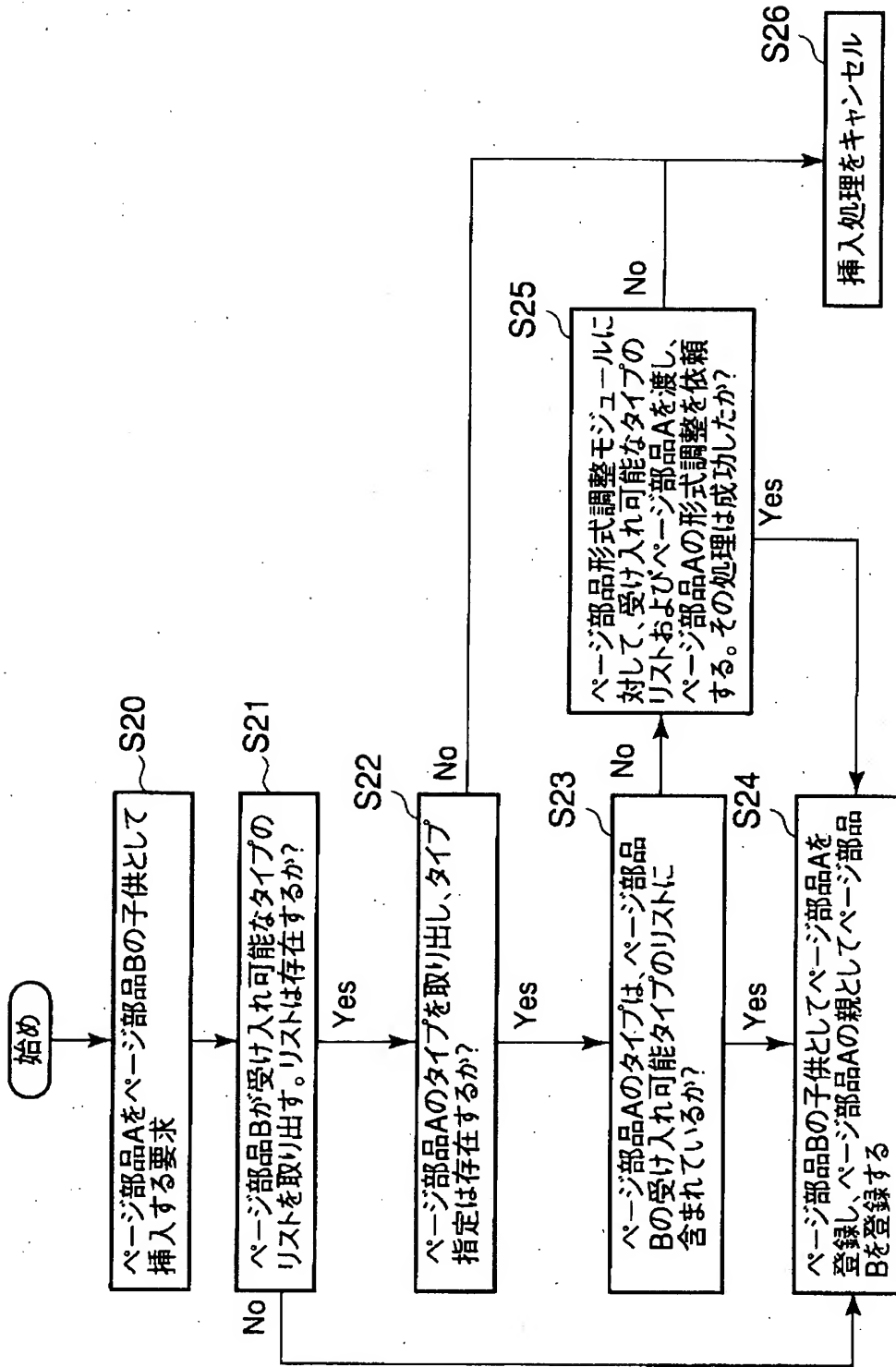
【図 9】



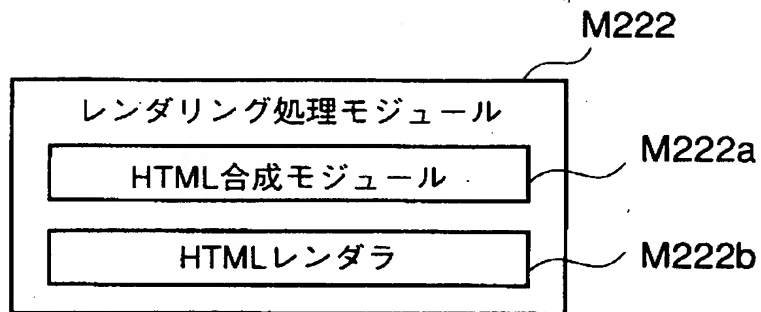
【図10】



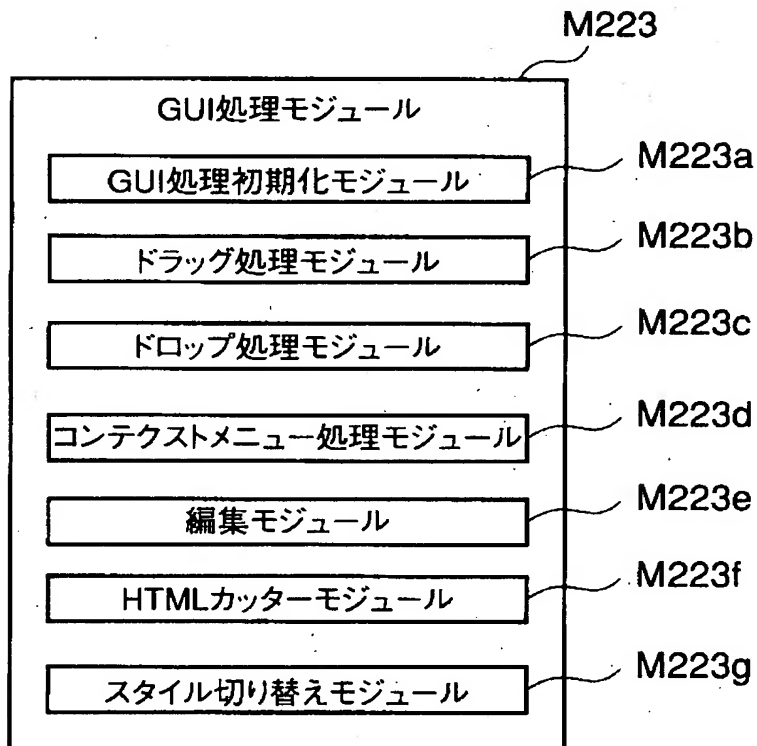
【図11】



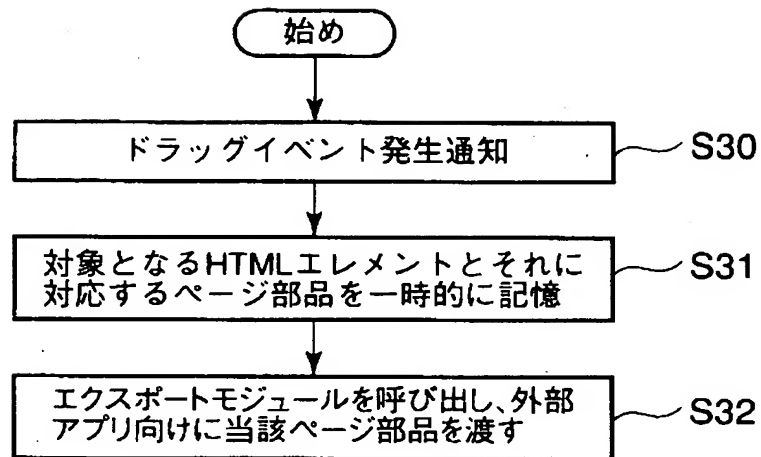
【図12】



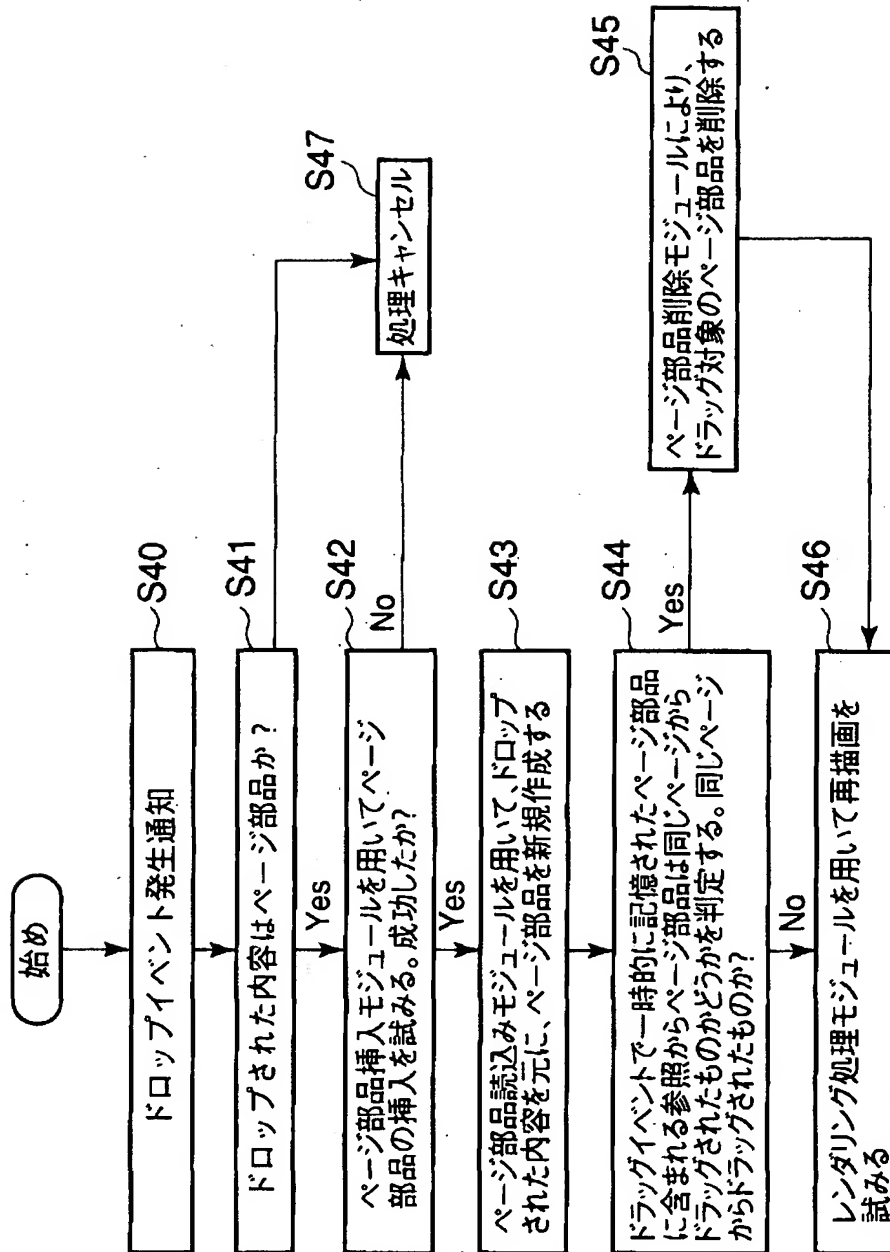
【図13】



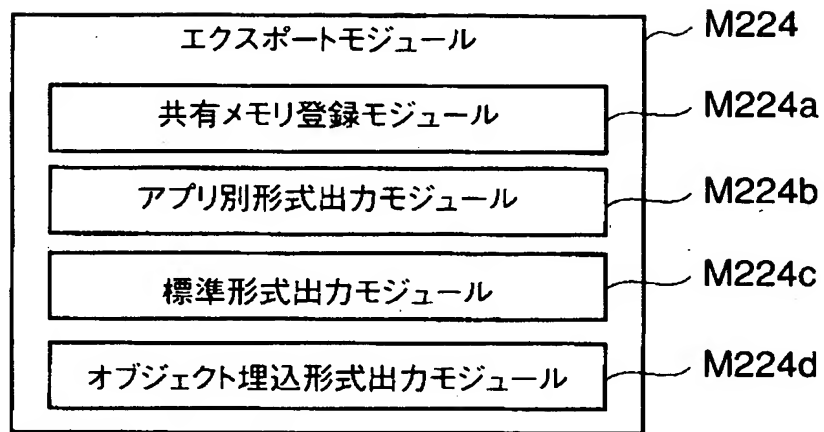
【図 14】



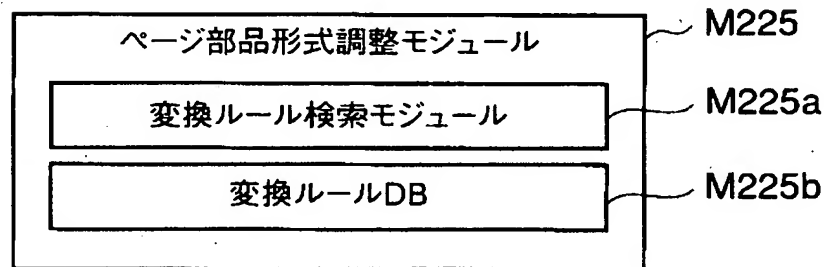
【図15】



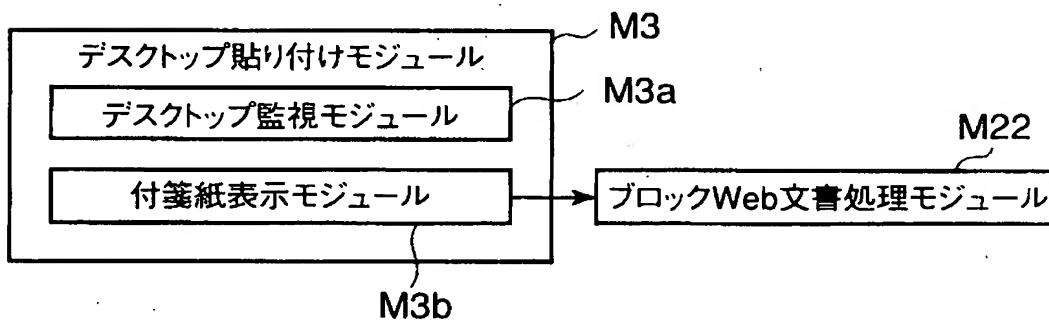
【図 16】



【図 17】

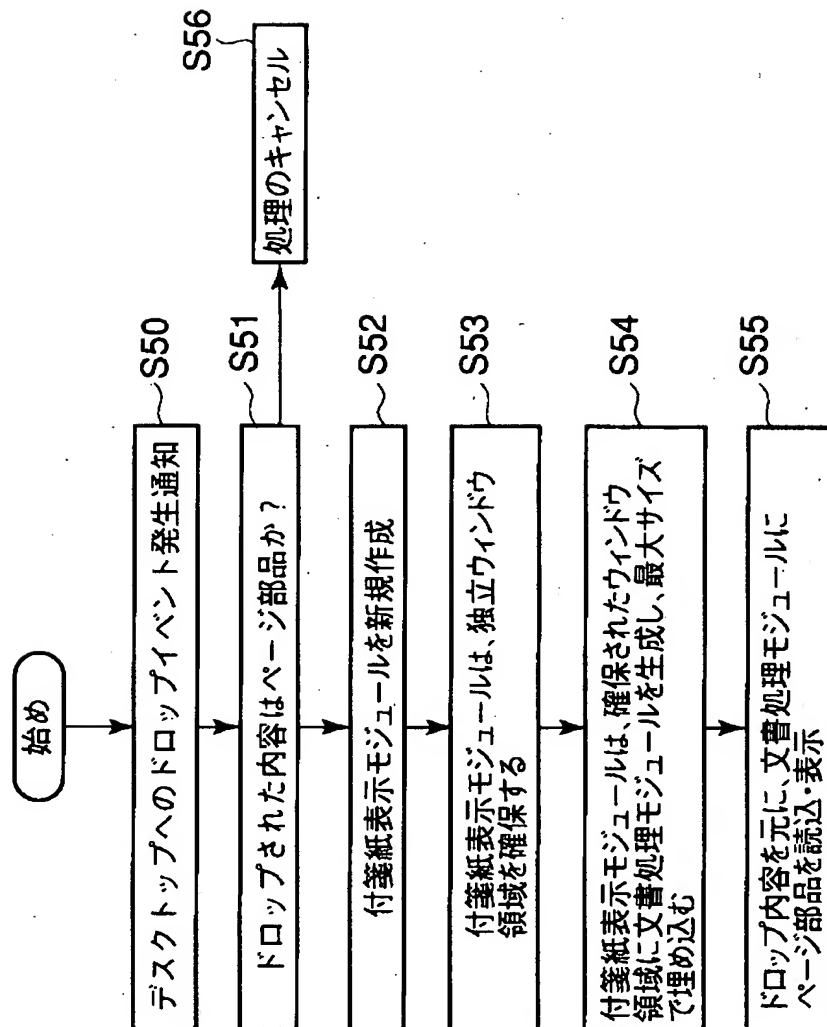


【図 18】

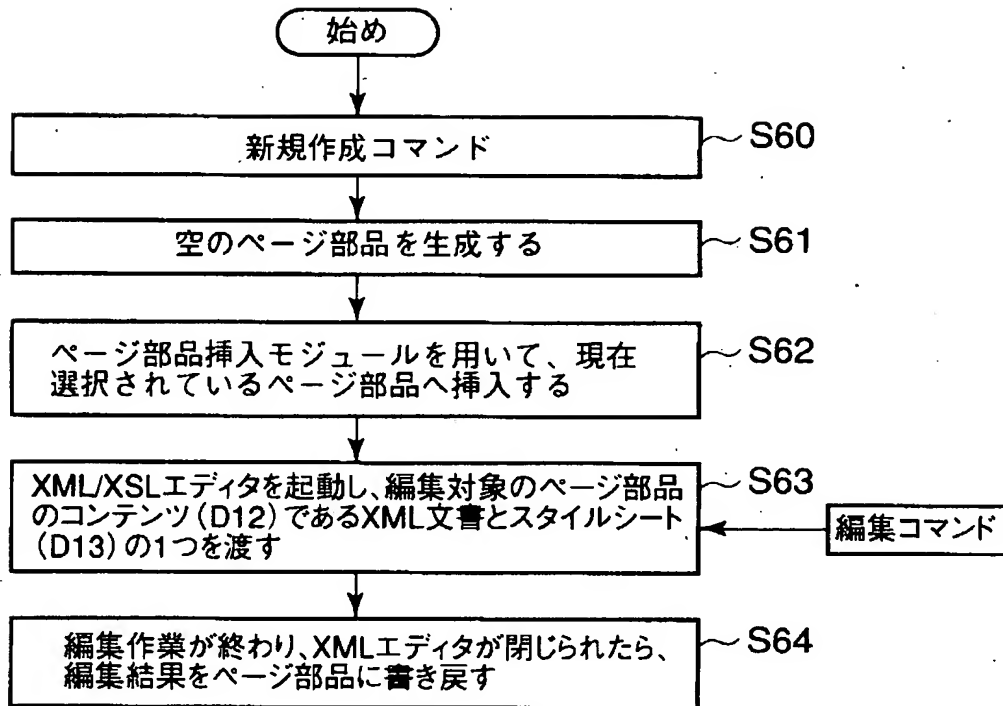




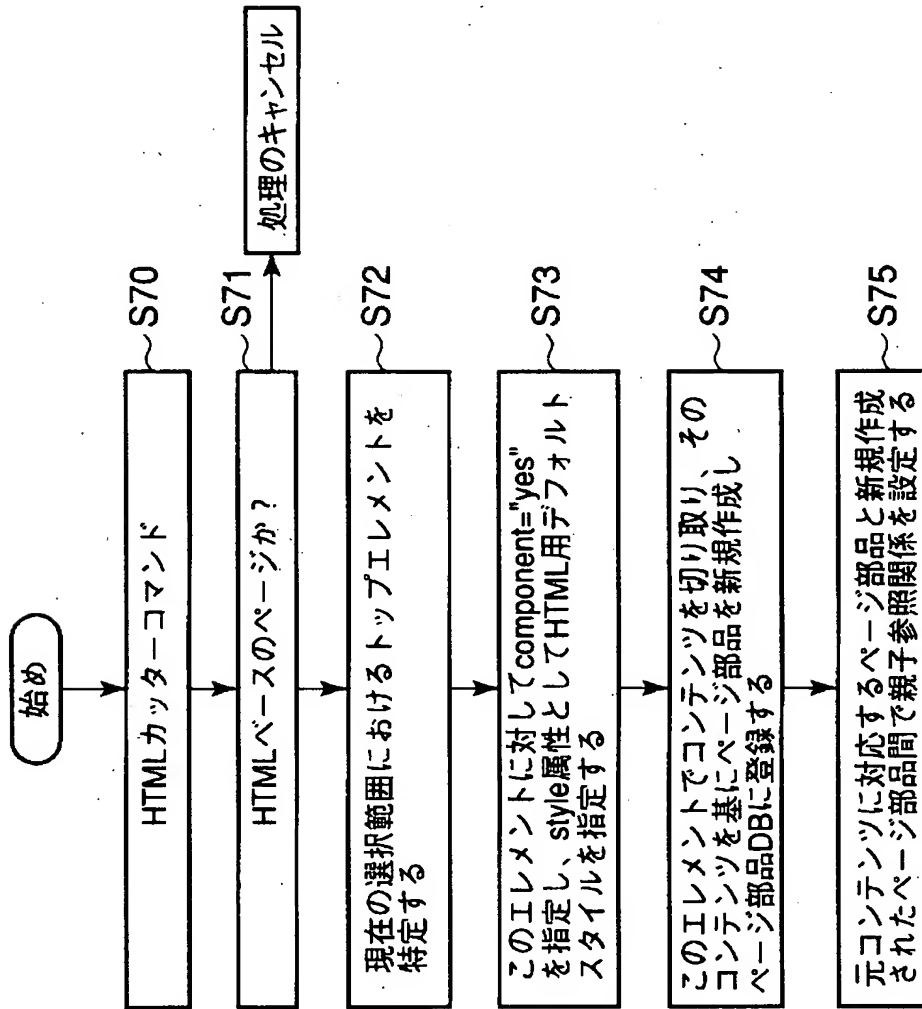
【図 19】



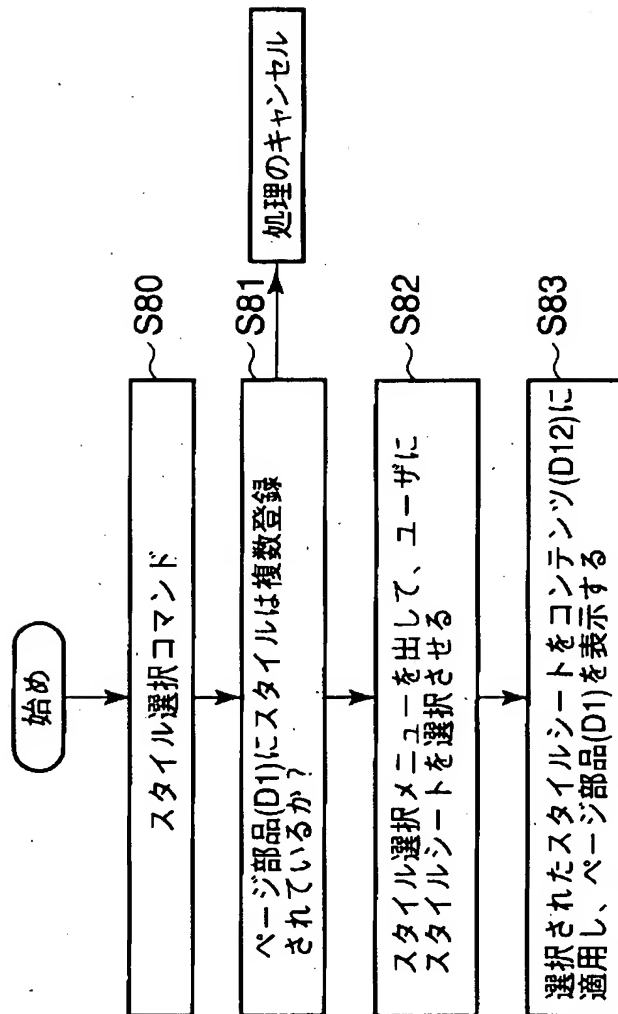
【図20】



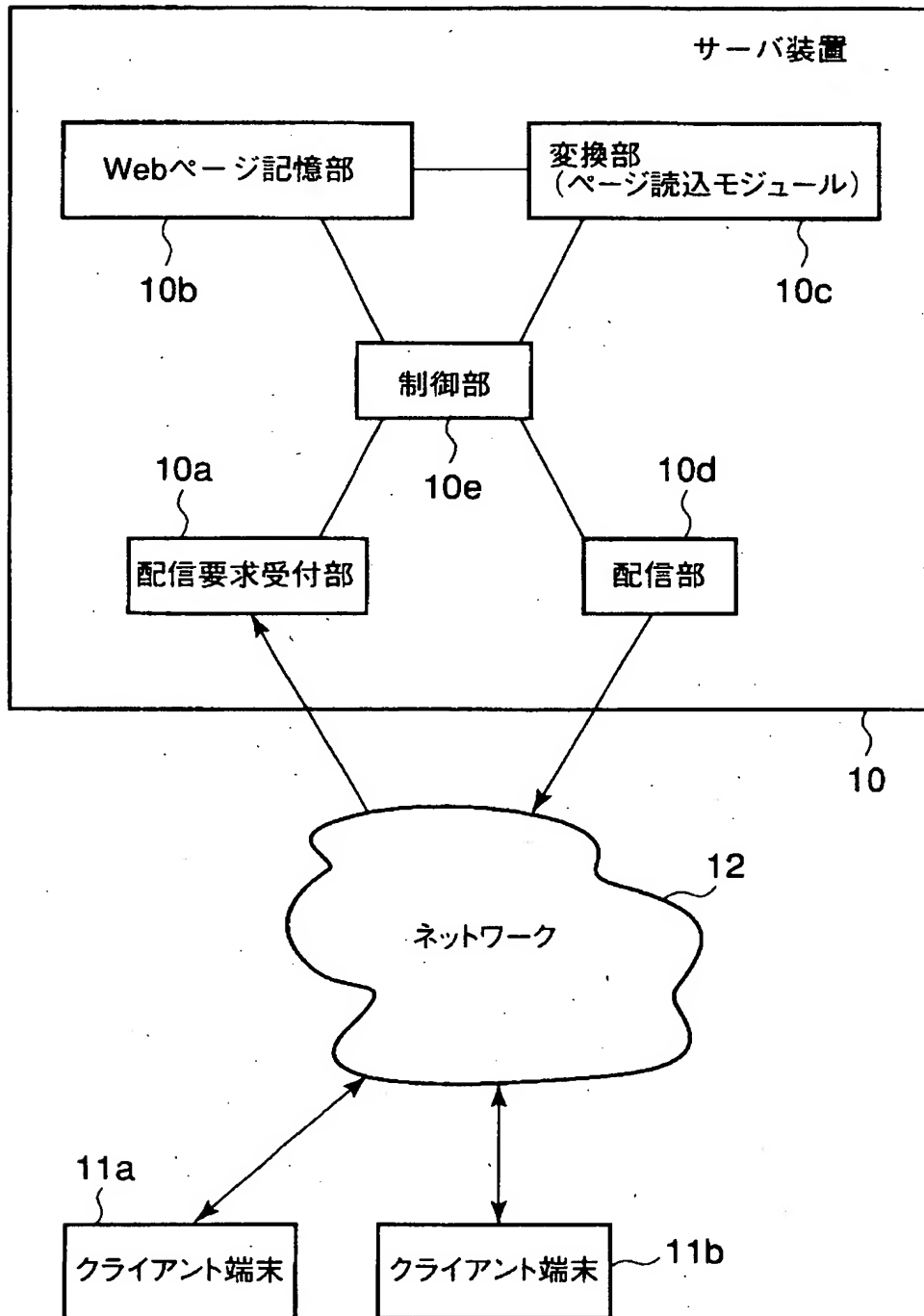
【図 21】



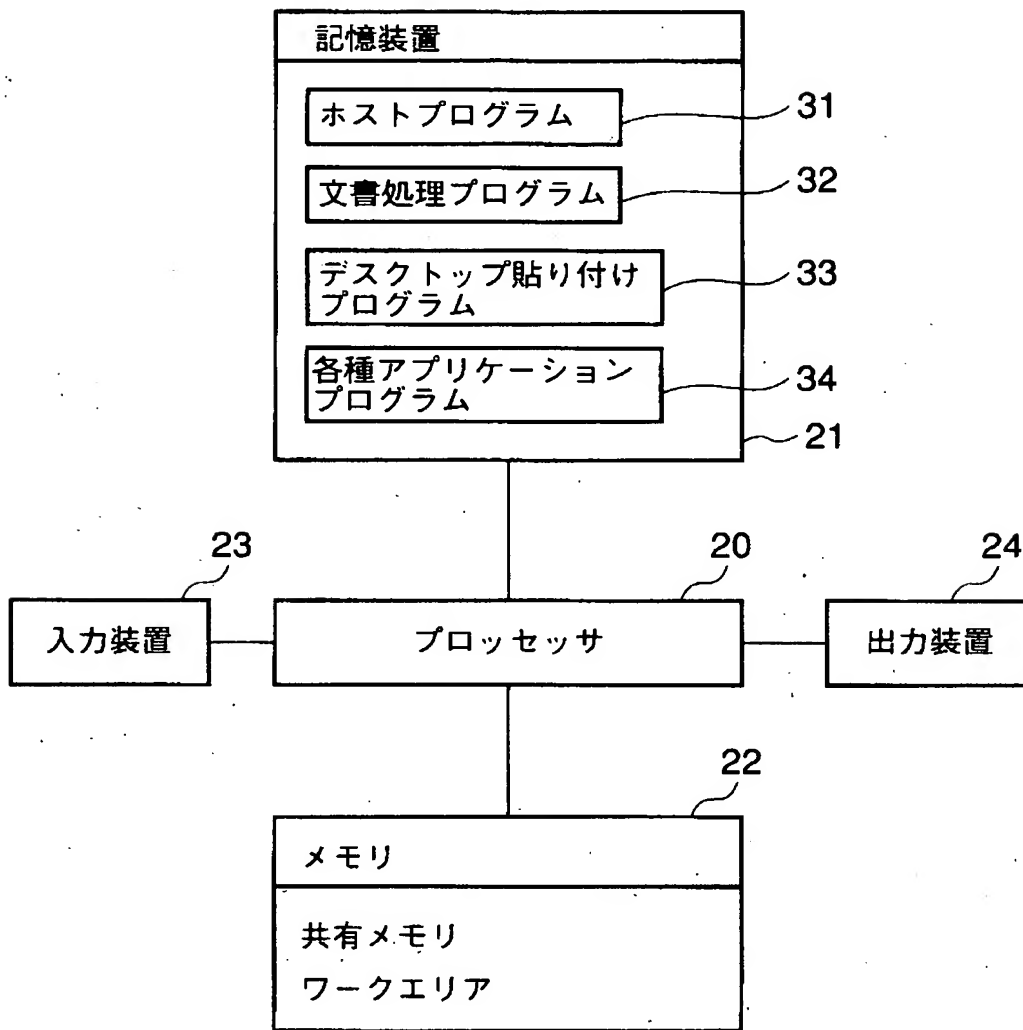
【図 22】



【図 23】



【図 24】



【書類名】 要約書

【要約】

【課題】構造化文書の中に含まれるコンテンツを、それよりさらに細かい操作単位としてのサブコンテンツ単位で容易に編集することができる、文書編集方法およびシステムを提供する。

【解決手段】構造化文書に一致または含まれる、少なくとも1つの要素からなる予め操作単位として定められた部分文書には、少なくとも当該部分文書が操作単位であることを表す情報が含まれており、この情報を基に、少なくとも当該部分文書と当該部分文書の位置情報とを含む部分文書毎の部品データを作成して、構造化文書を部品データの集合として記憶手段に記憶し、操作対象として選択された前記部分文書に対する操作内容に応じて、当該操作対象の部分文書に対応する部品データを更新することにより構造化文書を編集する。

【選択図】 図1

出 願 人 履 歴 情 報

識別番号 [000003078]

1. 変更年月日 2001年 7月 2日  
[変更理由] 住所変更  
住 所 東京都港区芝浦一丁目1番1号  
氏 名 株式会社東芝